Web Application Hacking Lesson Cross Request Site Forgery

Objectives

- Review lecture
- Complete lab on CRSF

Lab for CRSF

Lab Prep

Open the link to https://hack.me

Choose “Start a hackme”

Scroll down and select “DVWA 1.0.7”

Accept the agreement after selecting “anonymous login”

DVWA Cross Site Request Forgery

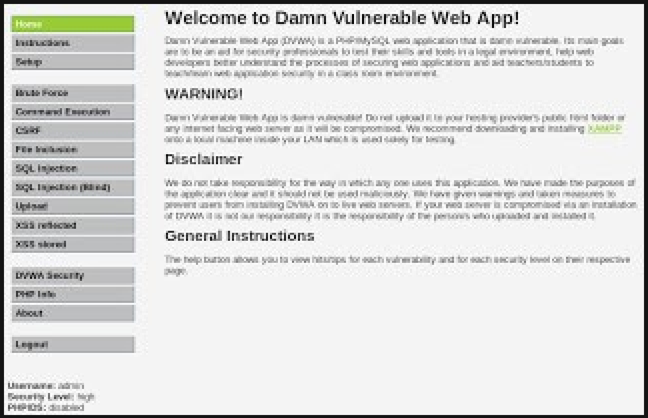
Cross Site Request Forgery is very dangerous, and also quite common. OWASP describes Cross Site Request Forgery as:

Cross-Site Request Forgery (CSRF) is an attack that tricks the victim into loading a page that contains a malicious request. It is malicious in the sense that it inherits the identity and privileges of the victim to perform an undesired function on the victim's behalf, like change the victim's e-mail address, home address, or password, or purchase something. CSRF attacks generally target functions that cause a state change on the server but can also be used to access sensitive data.

For most sites, browsers will automatically include with such requests any credentials associated with the site, such as the user's session cookie, basic auth credentials, IP address, Windows domain credentials, etc. Therefore, if the user is currently authenticated to the site, the site will have no way to distinguish this from a legitimate user request.

In this way, the attacker can make the victim perform actions that they didn't intend to, such as logout, purchase item, change account information, retrieve account information, or any other function provided by the vulnerable website.

In other words, this is one reason why people tell you not to click on links (or open e-mails) from people you do not trust. Just by clicking a link they can steal your information or change your password without you knowing about it.



Welcome to Damn Vulnerable Web App!

Damn Vulnerable Web App (DVWA) is a PHP/MySQL web application that is damn vulnerable. Its main goals are to be an aid for security professionals to test their skills and tools in a legal environment, help web developers better understand the processes of securing web applications and aid teachers/students to teach/learn web application security in a class room environment.

WARNING!

Damn Vulnerable Web App is damn vulnerable! Do not upload it to your hosting provider's public html folder or any internet facing web server as it will be compromised. We recommend downloading and installing XAMPP onto a local machine inside your LAN which is used solely for testing.

Disclaimer

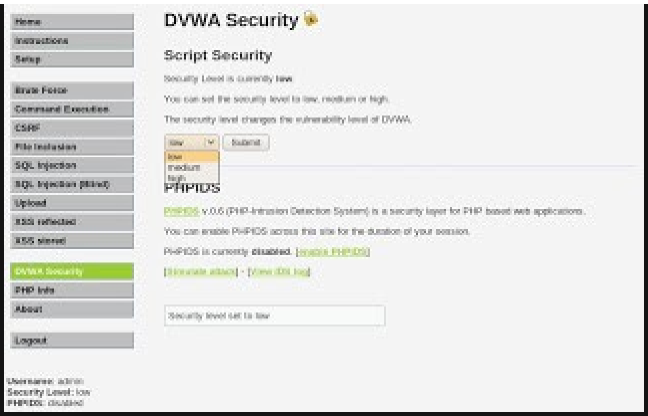
We do not take responsibility for the way in which any one uses this application. We have made the purposes of the application clear and it should not be used maliciously. We have given warnings and taken measures to prevent users from installing DVWA on to live web servers. If your web server is compromised via an installation of DVWA it is not our responsibility it is the responsibility of the person/s who uploaded and installed it.

General Instructions

The help button allows you to view hits/tips for each vulnerability and for each security level on their respective page.

Username: admin
Security Level: high
PHPIDS: disabled

2. **Go to the DVWA Security** page and **change the Script Security level to low**.



DVWA Security

Script Security

Security Level is currently **low**.

You can set the security level to low, medium or high.

The security level changes the vulnerability level of DVWA.

PHPIDS

PHPIDS v.0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.

You can enable PHPIDS across this site for the duration of your session.

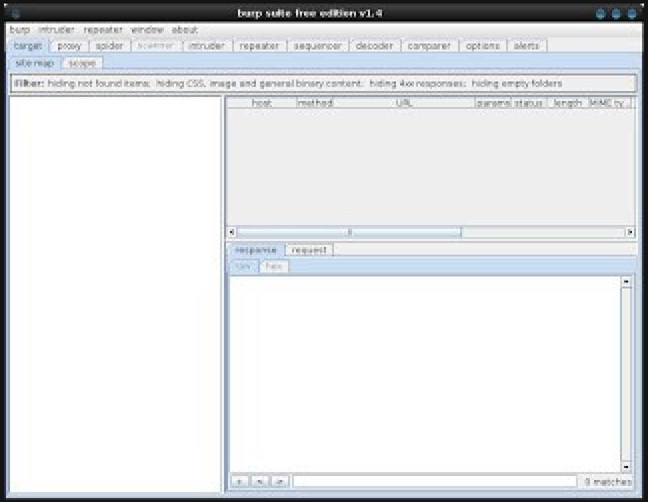
PHPIDS is currently **disabled**. [enable PHPIDS]

[Simulate attack] - [View IDS log]

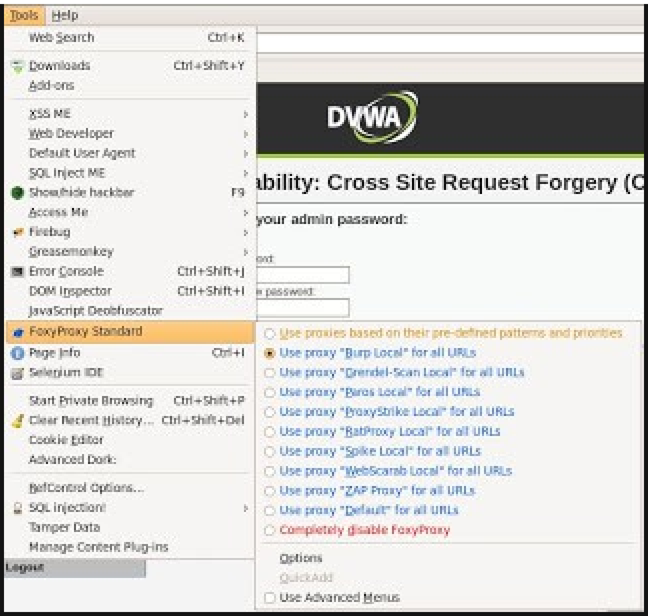
Security level set to low

Username: admin
Security Level: low
PHPIDS: disabled

3. **Open Burp Suite** by going to Backtrack **> Vulnerability Assesment> Web Application Assessment->Web Application Proxies > BurpSuite** and set **Firefox** to use the **Burp Local Proxy** by clicking **Tools > FoxyProxy Standard > Use proxy "<whatever you set for Zap>" for all URLs**.



Filter: hiding not found items; hiding CSS, image and general binary content; hiding 4xx responses; hiding empty folders

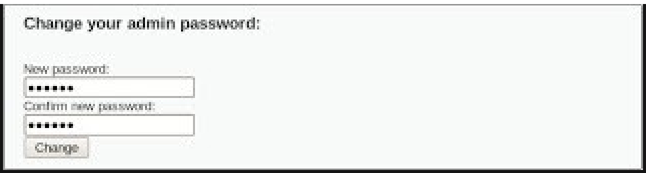

Tools
Help
Web Search Ctrl+K
Downloads Ctrl+Shift+Y
Add-ons
XSS ME
Web Developer
Default User Agent
SQL Inject ME
Show/hide hackbar F9
Access Me
Firebug
Greasemonkey
Error Console Ctrl+Shift+J
DOM Inspector Ctrl+Shift+I
JavaScript Deobfuscator
FoxyProxy Standard
Page Info Ctrl+I
Selenium IDE
Start Private Browsing Ctrl+Shift+P
Clear Recent History... Ctrl+Shift+Del
Cookie Editor
Advanced Dork:
RefControl Options...
SQL Injection!
Tamper Data
Manage Content Plug-ins
Logout
DVWA
ability: Cross Site Request Forgery (C
your admin password:
Use proxies based on their pre-defined patterns and priorities
Use proxy "Burp Local" for all URLs
Use proxy "Grendel-Scan Local" for all URLs
Use proxy "Paros Local" for all URLs
Use proxy "ProxyStrike Local" for all URLs
Use proxy "RatProxy Local" for all URLs
Use proxy "Spike Local" for all URLs
Use proxy "WebScarab Local" for all URLs
Use proxy "ZAP Proxy" for all URLs
Use proxy "Default" for all URLs
Completely disable FoxyProxy
Options
QuickAdd
Use Advanced Menus

Now we are ready to get started.

The Attack:

We are going to use Burp Suite to capture the HTTP request when we try to change a password, and from that we will create a "hidden" link to send to a victim (via e-mail, IM, etc.) that will change their password to whatever we want so we can access their account.

1. **Go to the CSRF** page in DVWA and **Change your admin password** by entering a password in the **New password** and **Confirm new password** fields and clicking the **Change** button.

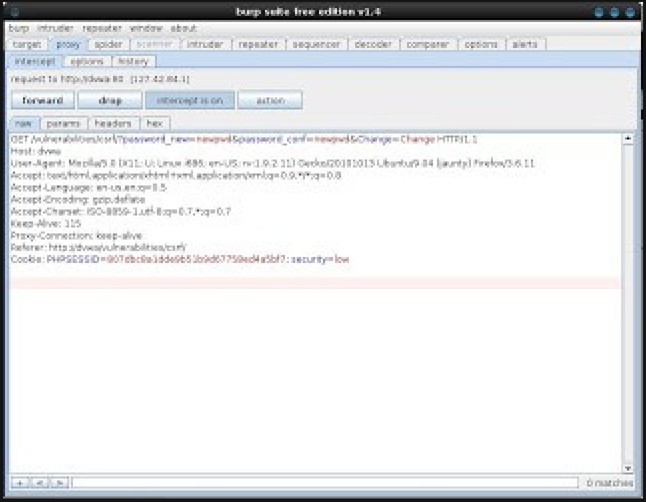


Change your admin password:

New password:
••••••
Confirm new password:
••••••

Notice that the page is loading, but not complete. That is because we need to tell Burp Suite to forward the packet and let it finish it's process.

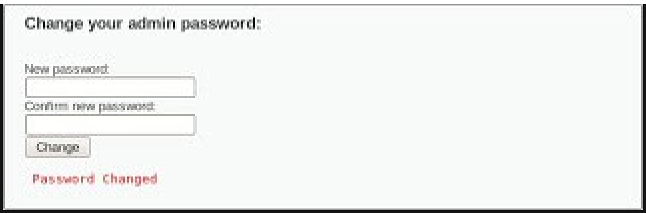
2. **Go to Burp Suite**, click the **Proxy** tab, and view the password change http request and forward it after and you will see that your Password Changed on the DVWA site.



burp suite free edition v1.4
burp intruder repeater window about
target proxy spider scanner intruder repeater sequencer decoder comparer options alerts
intercept options history
forward drop intercept is on action
raw params headers hex
0 matches

request to http://dvwa:80 [127.42.84.1]

```
GET /vulnerabilities/csrf/?password_new=newpwd&password_conf=newpwd&Change=Change HTTP/1.1
Host: dvwa
User-Agent: Mozilla/5.0 (X11; U; Linux i686; en-US; rv:1.9.2.11) Gecko/20101013 Ubuntu/9.04 (jaunty) Firefox/3.6.11
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
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 115
Proxy-Connection: keep-alive
Referer: http://dvwa/vulnerabilities/csrf/
Cookie: PHPSESSID=907dbc0a1dde9b51b9d67759ed4a5bf7; security=low
```



Password Changed

Now the part we are interested in is the begenning of the http request which looks something like:

```
GET /dvwa/vulnerabilities/csrf/?password_new=newpwd&password_conf=newpwd&Change=Change HTTP/1.1
```

Now all we have to do is construct a link that will perform the same function and hide it in some html so our victim doesn't know it is happening.

Example Text Link:

```
<a href="http://dvwa/vulnerabilities/csrf/?password_new=admin&password_conf=admin&Change=Change">Click
Here</a>
```

Looks like: Click Here

Now, obviously, we can be more creative than a "Click Here" link but I'll leave that part up to you.

When someone clicks on the "Click Here" link it will change their password to "admin" without them knowing about it. Feel free to try it out and see for yourself. Now imagine this on a bank website, where all you needed to do was get someone to click a link that would deposit X amount of money in Y account, or on Facebook where you could change peoples passwords and get into all the hidden stuff they locked you out of.

1. Describe in your own words CRSF?
2. What OWASP ranking is CRSF?
3. Google and explain some of the dangers that CRSF poses?

  - **Proof of Lab Instructions**:
    1. Do a <PrtScn> of lab
    2. Paste into a word document
    3. Post to teambox
  -