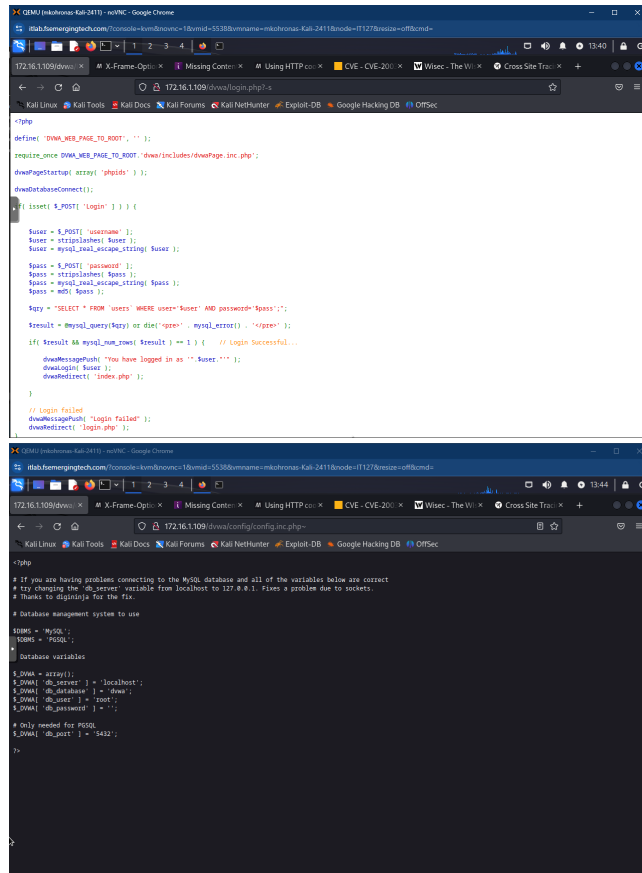


## Main Objective: Highlight all sensitive information



```
<?php
define( 'DWA_WEB_PAGE_TO_ROOT', '' );
require_once DWA_WEB_PAGE_TO_ROOT . 'dwa/includes/dwaPage.inc.php';

dwaPageStartup( array( 'phpinfo' ) );

dwaDatabaseConnect();

if (isset( $_POST['login'] ) ) {

    $user = $_POST['username'];
    $user = stripslashes( $user );
    $user = mysql_real_escape_string( $user );

    $pass = $_POST['password'];
    $pass = stripslashes( $pass );
    $pass = mysql_real_escape_string( $pass );
    $pass = md5( $pass );

    $sql = "SELECT * FROM 'users' WHERE user='$user' AND password='$pass'";
    $result = mysql_query($sql) or die( 'error' . mysql_error() . "<br>");
    if ( $result && mysql_num_rows( $result ) == 1 ) { // Login Successful...
        dwaMessagePush( "You have logged in as '$user.'" );
        dwaLogin( $user );
        dwaRedirect( 'index.php' );
    }

    // Login failed
    dwaMessagePush( "Login failed" );
    dwaRedirect( 'login.php' );
}


```

```
<?php
# If you are having problems connecting to the MySQL database and all of the variables below are correct
# try changing the db_server variable from localhost to 127.0.0.1. Fixes a problem due to sockets.
# Thanks to digitalia for the fix.

# Database management system to use
$dbms = 'mysql';
$pass = 'root';

# Database variables
$dbwa = array();
$dbwa['db_server'] = 'localhost';
$dbwa['db_database'] = 'dwa';
$dbwa['db_user'] = 'root';
$dbwa['db_password'] = '';

# Only needed for MySQL
$dbwa['db_port'] = '3306';

```

Both the login.php?-s file and the config.inc.php~ files are both leaking sensitive information about post calls and error messages. In addition to all the database information.

Ensuring we test each php file with a “~” is important as those are usually how IT teams mark backups which they may forget to backup.

Directories sensitive information found:

172.16.1.109/dvwa/login.php?-s

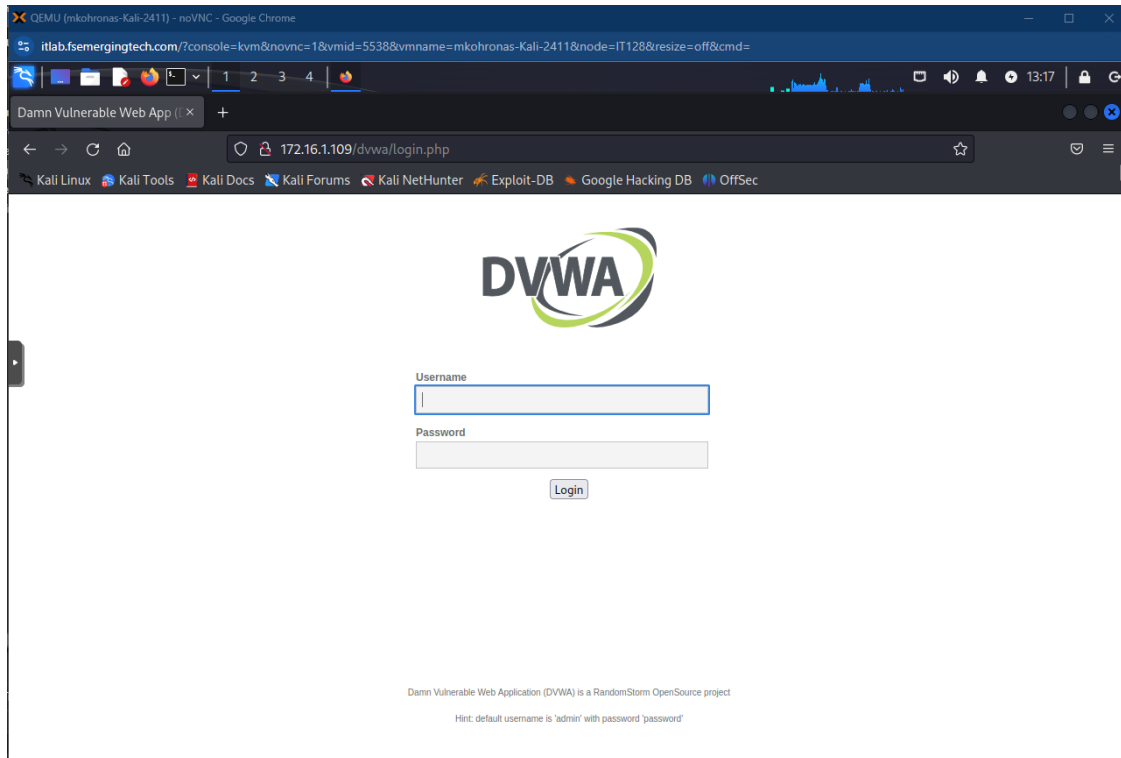
172.16.1.109/dvwa/config.inc.php~

Extra Information:

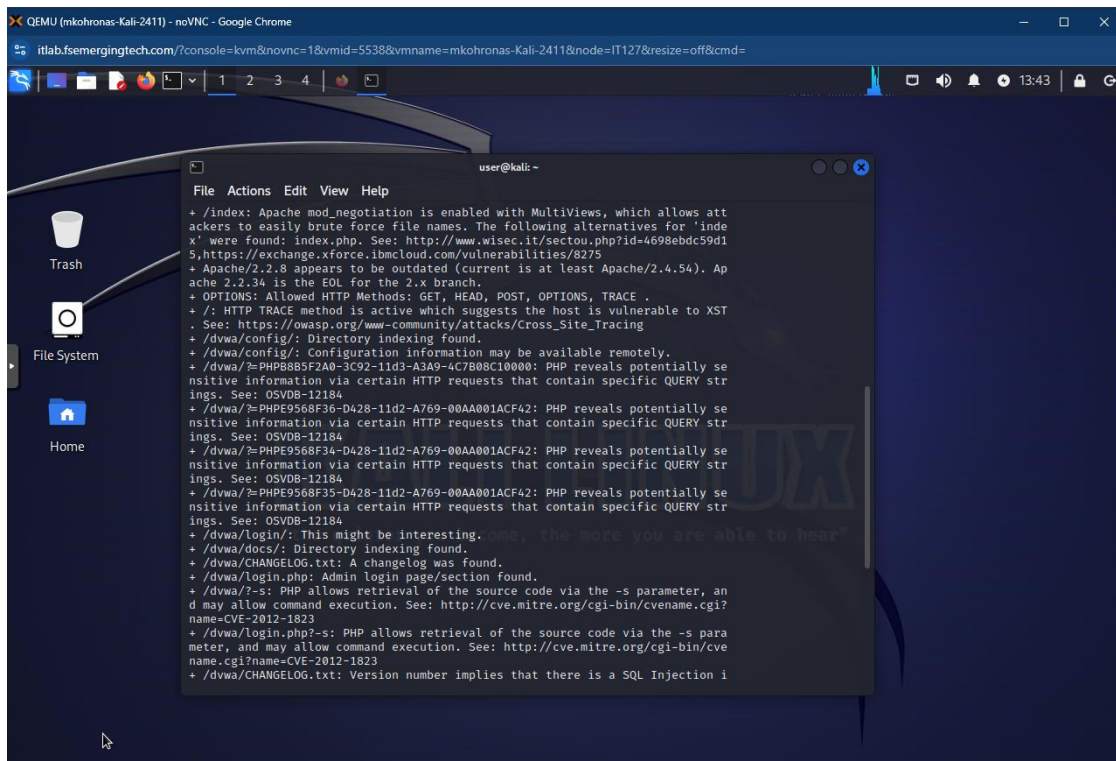
PHP Credits may provide version information which is beneficial for attackers

A change log can give a attacker an idea of what systems are being used

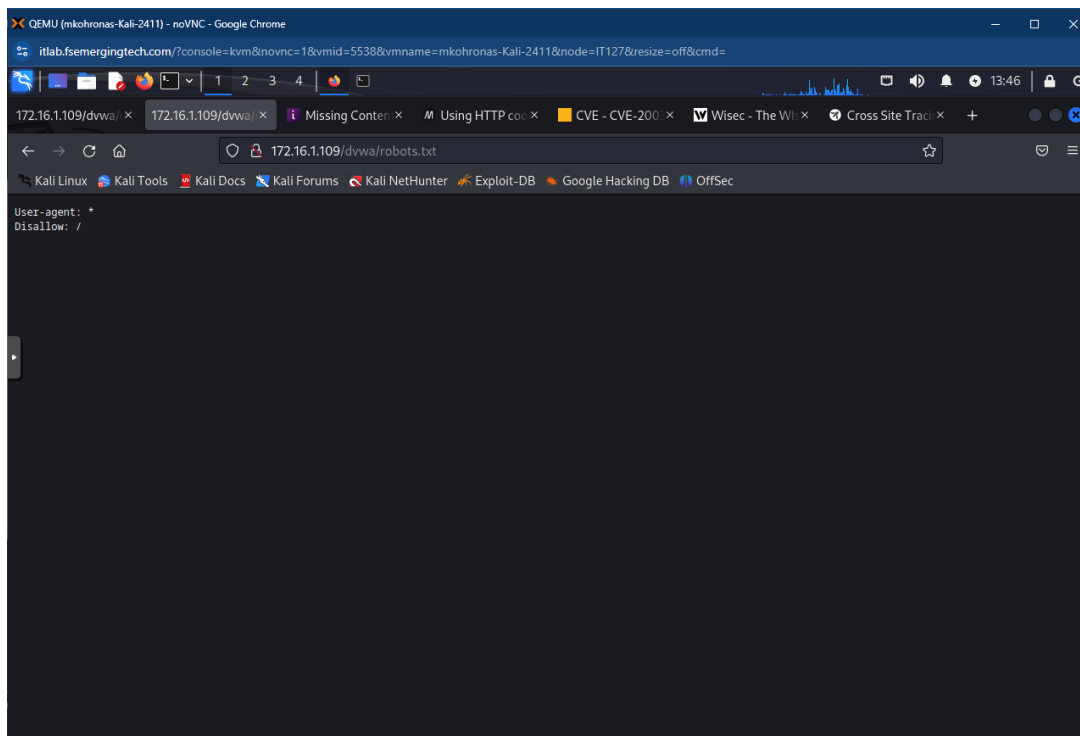
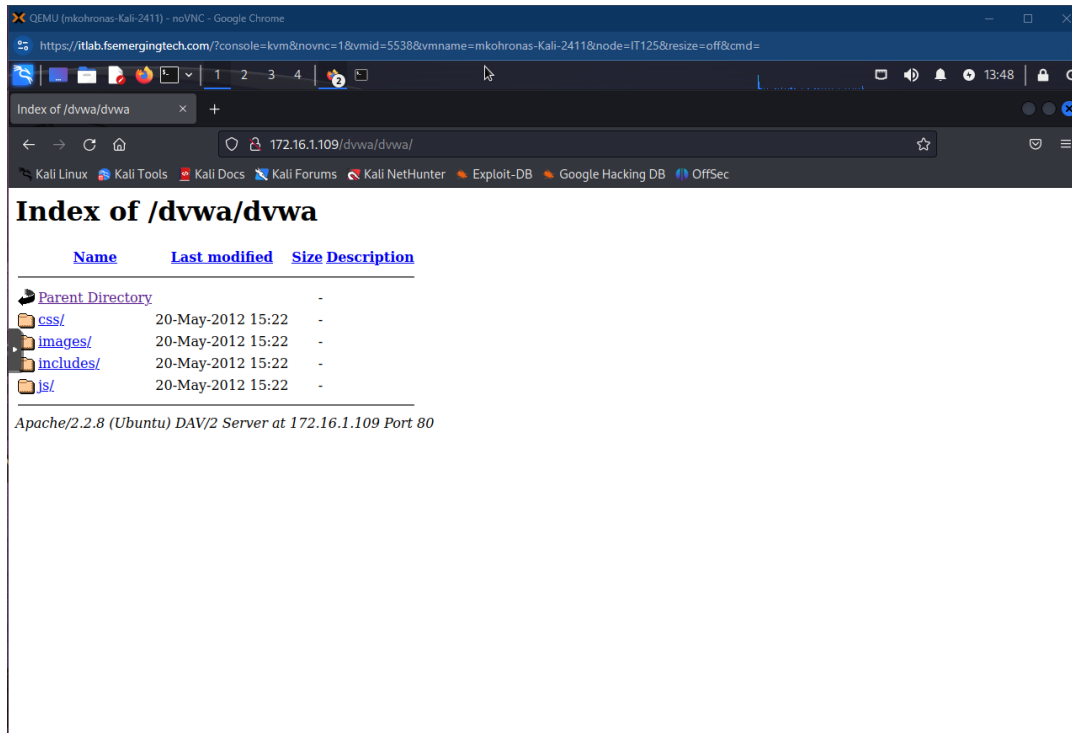
## Step 1: go to site and perform scan



## Step 2: perform scan



### Step 3: Search all directories



QEMU (mkohronas-Kali-2411) - noVNC - Google Chrome

itlab.fsemergingtech.com/?console=kvm&novnc=1&vmid=5538&vmname=mkohronas-Kali-2411&node=IT127&resize=off&cmd=

172.16.1.109/dvwa/ x phpinfo() x Missing Content x M Using HTTP coo x CVE - CVE-200 x Wisec - The Wi x Cross Site Traci x + Lock Screen

172.16.1.109/dvwa/?=PHPB8B5F2A0-3C92-11d3-A3A9-4C7B08C10000

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

## PHP Credits

PHP Group	
Thies C. Arntzen, Stig Bakken, Shane Caraveo, Andi Gutmans, Rasmus Lerdorf, Sam Ruby, Sascha Schumann, Zeev Suraski, Jim Winstead, Andrei Zmievski	

Language Design & Concept	
Andi Gutmans, Rasmus Lerdorf, Zeev Suraski	

PHP 5 Authors	
Contribution	Authors
Zend Scripting Language Engine	Andi Gutmans, Zeev Suraski
Extension Module API	Andi Gutmans, Zeev Suraski, Andrei Zmievski
UNIX Build and Modularization	Stig Bakken, Sascha Schumann
Win32 Port	Shane Caraveo, Zeev Suraski, Wez Furlong
Server API (SAPI) Abstraction Layer	Andi Gutmans, Shane Caraveo, Zeev Suraski
Streams Abstraction Layer	Wez Furlong, Sara Golemon
PHP Data Objects Layer	Wez Furlong, Marcus Boerger, Sterling Hughes, George Schlossnagle, Ilya Alshanetsky

SAPI Modules	
Contribution	Authors
AOLserver	Sascha Schumann
Apache 1.3 (apache_hooks)	Rasmus Lerdorf, Zeev Suraski, Stig Bakken, David Sklar, George Schlossnagle, Lukas Schroeder
Apache 1.3	Rasmus Lerdorf, Zeev Suraski, Stig Bakken, David Sklar
Apache 2.0 Filter	Sascha Schumann, Aaron Bannert
Apache 2.0 Handler	Ian Holmes, Justin Erenkrantz (based on Apache 2.0 Filter code)

QEMU (mkohronas-Kali-2411) - noVNC - Google Chrome


itlab.fsemergingtech.com/?console=kvm&novnc=1&vmid=5538&vmname=mkohronas-Kali-2411&node=IT127&resize=off&cmd=

DVWA-Document... x X-Frame-Optio... x Missing Content x M Using HTTP coo x CVE - CVE-200 x Wisec - The Wi x Cross Site Traci x +

172.16.1.109/dvwa/docs/DVWA-Documentation.pdf

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

1 of 17 Automatic Zoom



**Damn Vulnerable Web Application (DVWA)**

**Documentation**

```
QEMU (mkohronas-Kali-2411) - noVNC - Google Chrome
itlab.fsemmergingtech.com/?console=kvm&novnc=1&vmid=5538&vmname=mkohronas-Kali-2411&node=IT127&resize=off&cmd=
172.16.1.109//dwa/
172.16.1.109//dwa/CHANGELOG.txt

#####
##### DAMN VULNERABLE WEB APP #####
#####

#####
# Change Log v1.0.7 #
#####

e-designed the login page + made some other slight cosmetic changes. 06/06/2010 (ethicalhack3r)
started PostgreSQL implementation. 15/03/2010 (ethicalhack3r)
few small cosmetic changes. 15/03/2010 (ethicalhack3r)
Improved the help information and look. 15/03/2010 (ethicalhack3r)
Fixed a few bugs thanks to Digininja. 15/03/2010 (ethicalhack3r)
Show logged in username. 05/02/2010 (Jason Jones)
Added new info on RandomStorm. 04/02/2010 (ethicalhack3r)
Added 'SQL Injection (Blind)'. 04/02/2010 (ethicalhack3r)
Added official documentation. 21/11/2009 (ethicalhack3r)
Implemented view all source functionality. 16/10/2009 (tnacuk, craig, ethicalhack3r)

#####
# Change Log v1.0.6 #
#####

Fixed a bug where the logo would not show on first time use. 03/09/2009 (ethicalhack3r)
Removed 'current password' input box for low+med CSRF security. 03/09/2009 (ethicalhack3r)
Added an article which was written for OWA5P Turkey. 03/10/2009 (ethicalhack3r)
Added more troubleshooting information. 02/10/2009 (ethicalhack3r)
Stored XSS high now sanitises output. 02/10/2009 (ethicalhack3r)
Fixed a 'bug' in XSS stored low which made it not vulnerable. 02/10/2009 (ethicalhack3r)
Rewritten command execution high to use a whitelist. 30/09/09 (ethicalhack3r)
Fixed a command execution vulnerability in exec high. 17/09/09 (ethicalhack3r)
Added some troubleshooting info for PHP 5.2.6 in readme.txt. 17/09/09 (ethicalhack3r)
Added the upload directory to the upload help. 17/09/09 (ethicalhack3r)

#####
```

```
QEMU (mkohronas-Kali-2411) - noVNC - Chrome
itlab.fsemmergingtech.com/?console=kvm&novnc=1&vmid=5538&vmname=mkohronas-Kali-2411&node=IT127&resize=off&cmd=
172.16.1.109//dwa/
172.16.1.109//dwa/7-s

<?php
define( 'DVWA_WEB_PAGE_TO_ROOT', '' );

require_once DVWA_WEB_PAGE_TO_ROOT.'dvwa/includes/dvwaPage.inc.php';

dvwaPageStartup( array( 'authenticated', 'phpids' ) );

$page = dvwaPageNewGrab();
$page[ 'title' ] .= $page[ 'title_separator' ].'Welcome';
$page[ 'page_id' ] = 'home';
$page[ 'body' ] .= "

<div class=\"body_padded\">

    <h1>Welcome to Damn Vulnerable Web App!</h1>

    <p>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 students to teach/learn web application security in a class room environment.</p>

    <h2> WARNING! </h2>

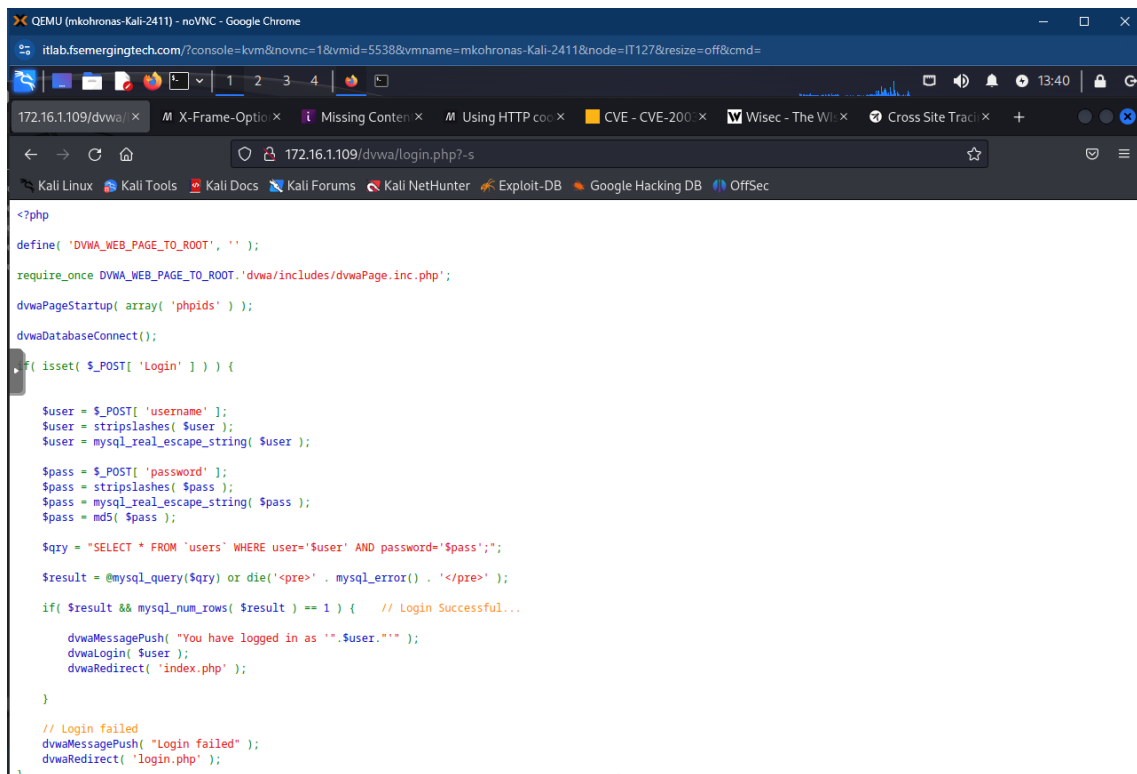
    <p>Damn Vulnerable Web App is damn vulnerable! Do not upload it to your hosting provider's public html folder or any internet facing web server as it will be compromise /en/xampp.html','XAMPP' )." onto a local machine inside your LAN which is used solely for testing.</p>

    <h2>Disclaimer</h2>

    <p>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</p>

    <h2>General Instructions</h2>

    <p>The help button allows you to view hits/tips for each vulnerability and for each security level on their respective page.</p>
</div>";
```



```
<?php

define( 'DVWA_WEB_PAGE_TO_ROOT', '' );

require_once DVWA_WEB_PAGE_TO_ROOT.'dwa/includes/dwaPage.inc.php';

dwaPageStartup( array( 'phpids' ) );

dwaDatabaseConnect();

if( isset( $_POST[ 'Login' ] ) ) {

    $user = $_POST[ 'username' ];
    $user = stripslashes( $user );
    $user = mysql_real_escape_string( $user );

    $pass = $_POST[ 'password' ];
    $pass = stripslashes( $pass );
    $pass = mysql_real_escape_string( $pass );
    $pass = md5( $pass );

    $qry = "SELECT * FROM 'users' WHERE user='$user' AND password='$pass'";

    $result = @mysql_query($qry) or die('<pre>'. mysql_error() . '</pre>');

    if( $result && mysql_num_rows( $result ) == 1 ) { // Login Successful...

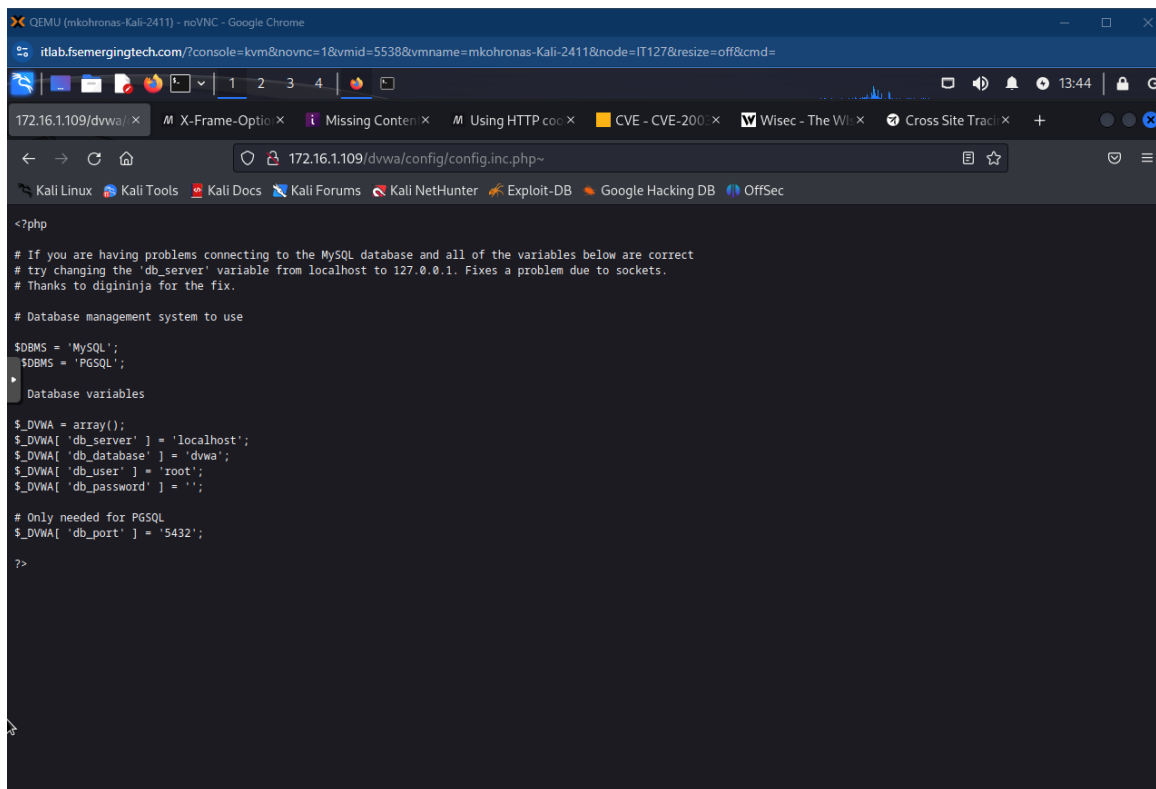
        dwaMessagePush( "You have logged in as '". $user . "' );
        dwaLogin( $user );
        dwaRedirect( 'index.php' );

    }

    // Login failed
    dwaMessagePush( "Login failed" );
    dwaRedirect( 'login.php' );

}
```

## Step 4: Test php files for ~ after them



```
<?php

# If you are having problems connecting to the MySQL database and all of the variables below are correct
# try changing the 'db_server' variable from localhost to 127.0.0.1. Fixes a problem due to sockets.
# Thanks to digininja for the fix.

# Database management system to use

$dbms = 'MySQL';
$dbms = 'PGSQL';

Database variables

$_DVWA = array();
$_DVWA[ 'db_server' ] = 'localhost';
$_DVWA[ 'db_database' ] = 'dvwa';
$_DVWA[ 'db_user' ] = 'root';
$_DVWA[ 'db_password' ] = '';

# Only needed for PGSQL
$_DVWA[ 'db_port' ] = '5432';

?>
```