

# DVWA安装及使用

---

## 安装

```
$ docker pull sagikazarmark/dvwa
```

## 启动镜像

```
$ docker run -d --name dvwa -p 8080:80 -p 33060:3306 sagikazarmark/dvwa
```

## 停止及删除容器

```
$ docker stop dvwa  
$ docker rm dvwa
```

## 环境

- PHP 5.6.30
- Apache 2.4.10
- MySQL 5.5.54

## 用户名/密码

- User: **root**
- Password: **password**
- Database: **dvwa**

## 初始化

访问 <http://IP:8080> 输入初始用户名密码

- User: **root**
- Password: **password**



Username

root

Password

.....

Login

You have logged out

点击 Create/Reset Database 进行初始化

← → 127.0.0.1/DVWA-1.9/setup.php

Setup DVWA

Instructions

About

## Database Setup

Click on the 'Create / Reset Database' button below to create or reset your database.  
If you get an error make sure you have the correct user credentials in: `D:\phpStudy\PHPTutorial\WWW\DVWA-1.9\config\config.inc.php`

If the database already exists, **it will be cleared and the data will be reset**.  
You can also use this to reset the administrator credentials ("**admin** / **password**") at any stage.

### Setup Check

Operating system: **Windows**  
Backend database: **MySQL**  
PHP version: **5.4.45**

Web Server SERVER\_NAME: **127.0.0.1**

PHP function display\_errors: **Enabled (Easy Mode!)**  
PHP function safe\_mode: **Disabled**  
PHP function allow\_url\_include: **Disabled**  
PHP function allow\_url\_fopen: **Enabled**  
PHP function magic\_quotes\_gpc: **Disabled**  
PHP module php-gd: **Installed**

reCAPTCHA key: **Missing**

Writable folder D:\phpStudy\PHPTutorial\WWW\DVWA-1.9\hackable/uploads/: **Yes**  
Writable file D:\phpStudy\PHPTutorial\WWW\DVWA-1.9\external\phpids\0.6/lib/IDS/tmp/phpids\_log.txt: **Yes**

**Status in red**, indicate there will be an issue when trying to complete some modules.

Create / Reset Database

[https://blog.csdn.net/qq\\_40023447](https://blog.csdn.net/qq_40023447)

## 重新登录页面

- User: **admin**
- Password: **password**

[Home](#)[Instructions](#)[Setup / Reset DB](#)[Brute Force](#)[Command Injection](#)[CSRF](#)[File Inclusion](#)[File Upload](#)[Insecure CAPTCHA](#)[SQL Injection](#)[SQL Injection \(Blind\)](#)[XSS \(Reflected\)](#)[XSS \(Stored\)](#)[DVWA Security](#)[PHP Info](#)[About](#)[Logout](#)

## Welcome to Damn Vulnerable Web Application!

Damn Vulnerable Web Application (DVWA) is a PHP/MySQL web application that is damn vulnerable. Its main goal is 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 to aid both students & teachers to learn about web application security in a controlled class room environment.

The aim of DVWA is to **practice some of the most common web vulnerability**, with **various difficulty levels**, with a simple straightforward interface.

### General Instructions

It is up to the user how they approach DVWA. Either by working through every module at a fixed level, or selecting any module and working up to reach the highest level they can before moving onto the next one. There is not a fixed object to complete a module; however users should feel that they have successfully exploited the system as best as they possible could by using that particular vulnerability.

Please note, there are **both documented and undocumented vulnerability** with this software. This is intentional. You are encouraged to try and discover as many issues as possible.

DVWA also includes a Web Application Firewall (WAF), PHPIDS, which can be enabled at any stage to further increase the difficulty. This will demonstrate how adding another layer of security may block certain malicious actions. Note, there are also various public methods at bypassing these protections (so this can be seen as an extension for more advance users)!

There is a help button at the bottom of each page, which allows you to view hints & tips for that vulnerability. There are also additional links for further background reading, which relates to that security issue.

### WARNING!

Damn Vulnerable Web Application is damn vulnerable! **Do not upload it to your hosting provider's public html folder or any Internet facing servers**, as they will be compromised. It is recommend using a virtual machine (such as [VirtualBox](#) or [VMware](#)), which is set to NAT networking mode. Inside a guest machine, you can download and install [XAMPP](#) for the web server and database.

### Disclaimer

选择项目难度，初始设置为Low

[Home](#)[Instructions](#)[Setup / Reset DB](#)[Brute Force](#)[Command Injection](#)[CSRF](#)[File Inclusion](#)[File Upload](#)[Insecure CAPTCHA](#)[SQL Injection](#)[SQL Injection \(Blind\)](#)[XSS \(Reflected\)](#)[XSS \(Stored\)](#)[DVWA Security](#)[PHP Info](#)[About](#)[Logout](#)

## DVWA Security

### Security Level

Security level is currently: **impossible**.

You can set the security level to low, medium, high or impossible. The security level changes the vulnerability level of DVWA:

1. Low - This security level is completely vulnerable and **has no security measures at all**. It's use is to be as an example of how web application vulnerabilities manifest through bad coding practices and to serve as a platform to teach or learn basic exploitation techniques.
2. Medium - This setting is mainly to give an example to the user of **bad security practices**, where the developer has tried but failed to secure an application. It also acts as a challenge to users to refine their exploitation techniques.
3. High - This option is an extension to the medium difficulty, with a mixture of **harder or alternative bad practices** to attempt to secure the code. The vulnerability may not allow the same extent of the exploitation, similar in various Capture The Flags (CTFs) competitions.
4. Impossible - This level should be **secure against all vulnerabilities**. It is used to compare the vulnerable source code to the secure source code.  
Priority to DVWA v1.9, this level was known as 'high'.

### PHPIDS

**PHPIDS** v0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.

PHPIDS works by filtering any user supplied input against a blacklist of potentially malicious code. It is used in DVWA to serve as a live example of how Web Application Firewalls (WAFs) can help improve security and in some cases how WAFs can be circumvented.

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

PHPIDS is currently: **disabled**. [[Enable PHPIDS](#)]