

WebGoat中文手册发布

傅奎 / i@isclab.org



OWASP 中国
The Open Web Application Security Project



- 简介

- Broken Web Application 系列
- Web安全漏洞实验环境
- 交互式教学环境
- 支持自定义实验内容



OWASP WebGoat v5.4

[Hints](#) [Show Params](#) [Show Cookies](#) [Lesson Plan](#) [Show Java](#) [Solution](#)

Command Injection

Introduction

General

Access Control Flaws

AJAX Security

Authentication Flaws

Buffer Overflows

Code Quality

Concurrency

Cross-Site Scripting (XSS)

Improper Error Handling

Injection Flaws

[Command Injection](#)[Numeric SQL Injection](#)[Log Spoofing](#)[XPath Injection](#)[String SQL Injection](#)[LAB: SQL Injection](#)[Stage 1: String SQL Injection](#)[Stage 2: Parameterized Query #1](#)[Stage 3: Numeric SQL Injection](#)[Stage 4: Parameterized Query #2](#)[Modify Data with SQL Injection](#)[Add Data with SQL Injection](#)[Database Backdoors](#)[Blind Numeric SQL Injection](#)[Blind String SQL Injection](#)

Denial of Service

Insecure Communication

Insecure Configuration

Insecure Storage

Malicious Execution

Parameter Tampering

Solution Videos

[Restart this Lesson](#)

Command injection attacks represent a serious threat to any parameter-driven site. The methods behind an attack are easy to learn and the damage caused can range from considerable to complete system compromise. Despite these risks an incredible number of systems on the internet are susceptible to this form of attack.

Not only is it a threat easily instigated, it is also a threat that, with a little common-sense and forethought, can be almost totally prevented. This lesson will show the student several examples of parameter injection.


It is always good practice to sanitize all input data, especially data that will be used in OS command, scripts, and database queries.

Try to inject a command to the operating system.

*** Congratulations. You have successfully completed this lesson.**

*** Congratulations. You have successfully completed this lesson.**

You are currently viewing: **BasicAuthentication.help**

Select the lesson plan to view: AccessControlMatrix.help  View

ExecResults for '/bin/sh'
Output...

Lesson Plan Title: Basic Authentication

Concept / Topic To Teach:

Basic Authentication is used to protect server side resources. The web server will send a 401 authentication request with the response for the requested resource. The client side browser will then prompt the user for a user name and password using a browser supplied dialog box. The browser will base64 encode the user name and password and send those credentials back to the web server. The web server will then validate the credentials and return the requested resource if



OWASP 中国

The Open Web Application Security Project

- WebGoat手册中文版

- 更新至5.4版本
- 目前是Beta版本
- 即将正式发布
- 20个技术章节
 - 概念
 - 原理
 - 目标
 - 方法

- ▣ 1 WebGoat简介

- 1.1 什么是WebGoat

- 1.2 什么是OWASP

- 1.3 WebGoat部署

- ▷ 1.4 用到的工具

- ▣ 2 WebGoat教程

- ▷ 2.1 综合 (General)

- ▷ 2.2 访问控制缺陷 (Access Control Flaws)

- ▷ 2.3 Ajax安全 (Ajax Security)

- ▷ 2.4 认证缺陷 (Authentication Flaws)

- ▷ 2.5 缓冲区溢出 (Buffer Overflows)

- ▷ 2.6 代码质量 (Code Quality)

- ▷ 2.7 并发 (Concurrency)

- ▷ 2.8 跨站脚本攻击 (Cross-Site Scripting (XSS))

- ▷ 2.9 不当的错误处理 (Improper Error Handling)

- ▷ 2.10 注入缺陷 (Injection Flaws)

- ▷ 2.11 拒绝服务 (Denial of Service)

- ▷ 2.12 不安全的通信 (Insecure Communication)

- ▷ 2.13 不安全的配置 (Insecure Configuration)

- ▷ 2.14 不安全的存储 (Insecure Storage)

- ▷ 2.15 恶意执行 (Malicious Execution)

- ▷ 2.16 参数篡改 (Parameter Tampering)

- ▷ 2.17 会话管理缺陷 (Session Management Flaws)

- ▷ 2.18 Web服务 (Web Services)

- ▷ 2.19 管理功能 (Admin Functions)

- ▷ 2.20 挑战 (Challenge)



- 价值

- 理解Web应用程序交互过程
- 理解漏洞产生的技术原理
- 掌握漏洞检测和修复的基本手段
- 提升技术人员的动手能力
- 提高Web安全防护水平



- 感谢项目组所有成员的努力
 - Rip、袁明坤、Tony、胡晓斌、beer、南国利剑、lion、蒋根伟、宋飞、蒋增、贺新朋、吴明、akast、杨天识、Snake、孟祥坤、范俊、阿保、Ivy、傅奎等（名不分先后）



OWASP 中国
The Open Web Application Security Project

Thanks

感谢所有关注并参与过OWASP项目的成员, 感谢你们的分享和付出, WebGoat和大家一起成长! 如有修改建议, 请发送至 webgoat@owasp.org.cn , 我们一起改进, 谢谢!