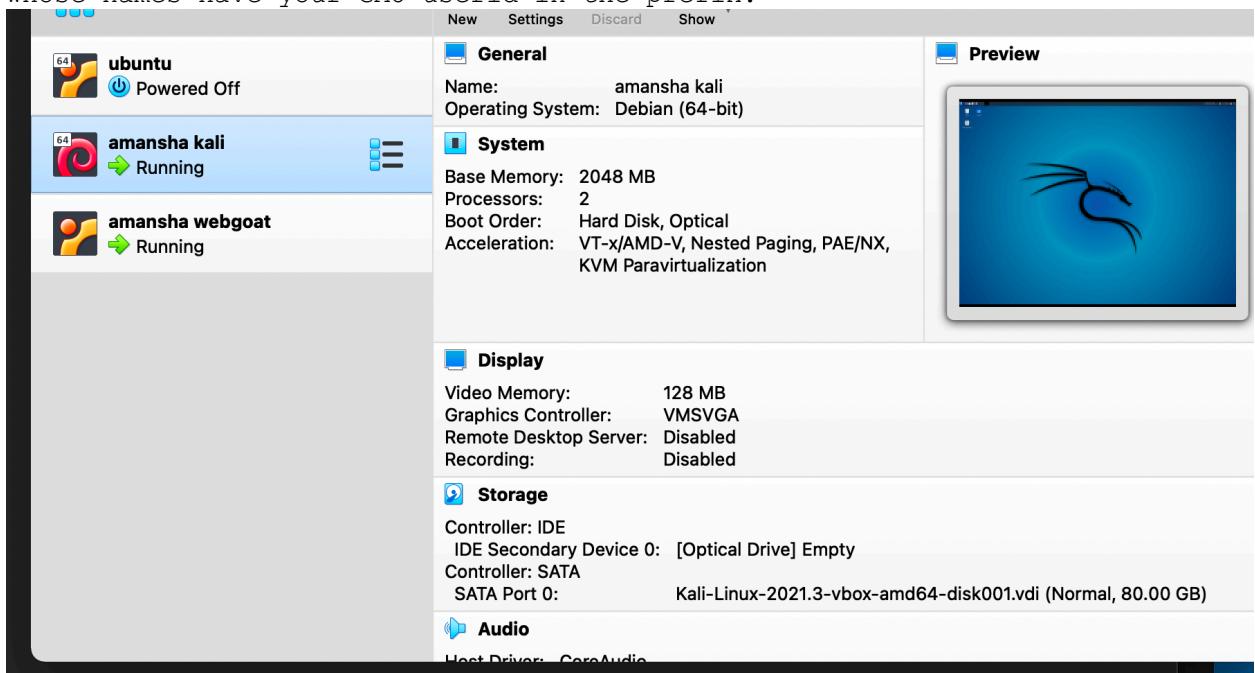


Amir Mansha

CYSE 450

Lab #6

- a. (2) Screenshot of VirtualBox showing your Kali and WebGoat machines whose names have your GMU userid in the prefix.



- b. (3) Screenshot showing the results in 3a above

The screenshot shows the OWASP WebGoat v5.4 application running in a Mozilla Firefox browser. The URL is `192.168.56.4/WebGoat/attack?Screen=20&menu=900`. The page title is **LAB: Cross Site Scripting**.

The sidebar on the left lists various security flaws:

- Introduction
- General
- Access Control Flaws
- AJAX Security
- Authentication Flaws
- Buffer Overflows
- Code Quality
- Concurrency
- Cross-Site Scripting (XSS)
- Improper Error Handling
- Injection Flaws
- Denial of Service
- Insecure Communication
- Insecure Configuration
- Insecure Storage
- Malicious Execution
- Parameter Tampering
- Session Management Flaws
- Web Services
- Admin Functions

The main content area shows the **Stage 6: Block Reflected XSS using Input Validation**. It includes a note: **THIS LESSON ONLY WORKS WITH THE DEVELOPER VERSION OF WEBGOAT**. A message box says: **Test stored XSS by your full name**. Below it, there's a form with fields: First Name: Tom Last Name: Cat Street: and a button: **Welcome Back Jerry**.

Amir Mansha

CYSE 450

Lab #6

- c. (3) Screenshot showing the results in 3b above

Introduction
General
Access Control Flaws
AJAX Security
Authentication Flaws
Buffer Overflows
Code Quality
Concurrency
Cross-Site Scripting (XSS)

[Phishing with XSS](#)

[LAB: Cross Site Scripting](#)

 [Stage 1: Stored XSS](#)

[Stage 2: Block Stored XSS using Input Validation](#)

[Stage 3: Stored XSS Revisited](#)

[Stage 4: Block Stored XSS using Output Encoding](#)

 [Stage 5: Reflected XSS](#)

[Stage 6: Block Reflected XSS](#)

Stored XSS Attacks

Reflected XSS Attacks

Cross Site Request Forgery (CSRF)

CSRF Prompt By-Pass

CSRF Token By-Pass

HTTPOnly Test

Cross Site Tracing (XST) Attacks

Improper Error Handling

Injection Flaws

Denial of Service

Insecure Communication

Insecure Configuration

Insecure Storage

Malicious Execution

Parameter Tampering

Session Management Flaws

Web Services

Admin Functions

Challenge

Solution Videos

Stage 6
Stage 6: Block Reflected XSS using Input Validation.

Editor. Check for Spelling, Grammar, and Writing Issues

THE DEVELOPER VERSION OF WEBGOAT

The XSS attack is no longer effective.

* You have completed Stage 5: Reflected XSS.
* Welcome to Stage 6: Block Reflected XSS



Amir Mansha

CYSE 450

Lab #6

- d. (4) Include your injection statement in the report. Screenshot showing results in 3c above.

[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](#)

[Session Management Flaws](#)

[Web Services](#)

[Admin Functions](#)

[Challenge](#)

It is always good practice to sanitize all input data, especially data that will be used in OS command, scripts, and database queries, even if the threat of SQL injection has been prevented in some other manner.

General Goal(s):

The form below allows a user to view their credit card numbers. Try to inject an SQL string that results in all the credit card numbers being displayed. Try the user name of 'Smith'.

* Congratulations. You have successfully completed this lesson.

* Now that you have successfully performed an SQL injection, try the same type of attack on a parameterized query. Restart the lesson if you wish to return to the injectable query.

Enter your last name:

SELECT * FROM user_data WHERE last_name = 'lim' OR '1' = '1'

USERID	FIRST_NAME	LAST_NAME	CC_NUMBER	CC_TYPE	COOKIE	LOGIN_COUNT
101	joe	Snow	987654321	VISA		0
101	joe	Snow	2234200065411	MC		0
102	John	Smith	2435600002222	MC		0
102	John	Smith	4352209902222	AMEX		0
103	Jane	Plane	123456789	MC		0
103	Jane	Plane	333498703333	AMEX		0
10312	Jolly	Hershey	176896789	MC		0
10312	Jolly	Hershey	333300003333	AMEX		0
10323	Grumpy	youaretheweakestlink	673834489	MC		0
10323	Grumpy	youaretheweakestlink	33413003333	AMEX		0
15603	Peter	Sand	123609789	MC		0
15603	Peter	Sand	338893453333	AMEX		0
15613	Joesph	Something	33843453533	AMEX		0

OWASP Foundation | Project WebGoat | Report Bug

- e. (4) Include your injection statement in the report. Screenshot results in 3d above.

Enter your userid: **jsmith**

USERID	SALARY
jsmith	20000

OWASP WebGoat v5.4 [Show Params](#) [Show Cookies](#) [Lesson Plan](#)

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](#)

Solution Videos

The form below allows a user to view salaries associated with a userid (from **salaries**). This form is vulnerable to String SQL Injection. In order to pass the injection to modify the salary for userid **jsmith**.

Enter your userid: **jsmith**

USERID **SALARY**

jsmith	70000
--------	-------

Created by Chuck Willis 
OWASP Foundation | Project WebGoat | Report Bug

Amir Mansha

CYSE 450

Lab #6

- f. (4) Include your injection statement in the report. Screenshot showing results in 3e above.

Solution Videos

The form below allows a user to view salaries. This form is vulnerable to String Injection to add a record to the table.

Enter your userid:

USERID	SALARY
Ismith	45000
wgoat	100000
rjones	777777
manderson	65000
jsmith	70000
amansha	30000

Amir Mansha
CYSE 450
Lab #6

g. Screenshot obtained in 4 above.

