PUNE INSTITUTE OF COMPUTER TECHNOLOGY,
DHANKAWADI PUNE-43.

A Project Report

On

Prevention of Web Applications from SQL Injection and Cross Site Scripting Attacks

SUBMITTED BY

Abhishek Sonawane 4174

CLASS: BE-1

COMPUTER ENGINEERING DEPARTMENT

Academic Year: 2019-20

1 Problem Statement

SQL Injection attacks and Cross -Site Scripting attacks are the two most common attacks on web applications. Develop a new policy based Proxy Agent, which classifies the request as a scripted request or query based request, and then, detects the respective type of attack, if any in the request. It should detect both SQL injection attacks as well as the Cross-Site Scripting attacks.

2 Abstract

A basic web application for banks was developed which was vulnerable to XSS and SQLi attacks. Later a policy based proxy agent was developed using Servlets and Filters in Java which can classify requests from clients as either scripted request or query based request.

This agent helps in detecting and mitigating the two types of attacks. The client is not aware about the presence of a proxy server for detecting attacks.

3 Objectives

1) Learn about SQL Injection(SQLi) and Cross-site Scripting(XSS) attacks.
2) Understand how these attacks can be detected and mitigated. 3) Learn about different types of SQLi and XSS attacks.
4) Build a web application that is vulnerable to SQLi and XSS. 5) Develop a policy based proxy agent to prevent these attacks. 6) Test the ability of web application to resist SQLi and XSS.

4 Hardware and Software Requirements

1) Ubuntu 18.04
2) Eclipse IDE for Java and Java EE 2020-03
3) JDK 1.8
4) Web browser Google Chrome and Firefox.
5) MySQL
6) JDBC jar

5 Introduction

In this era where the internet has captured the world, but the level of security that this internet provides has not grown as fast as the internet application. The Internet has eased the life of humans in numerous ways, but the drawbacks like the intrusions that are attached with the internet applications sustains the growth of these applications. One such intrusion is the SQL Injection Attacks (SQLIA). Since SQLIA contributes 25% of the total internet attacks, much research is being carried out in this area.

SQL injection vulnerabilities have been described as one of the most serious threats to database driven applications. Web applications that are vulnerable to SQL injection may allow an attacker to gain complete access to their underlying databases. A SQL Injection Attack usually starts with identifying weaknesses in the applications where unchecked user's input is transformed into database queries.

Code Injection is a type of attack in a web application, in which the attackers inject or provide some malicious code in the input data field to gain unauthorized and unlimited access, or to steal credentials from the users account. The injected malicious code executes as a part of the application. This results in either damage to the database, or an undesirable operation on the internet.

Reverse Proxy is a technique which is used to sanitize the user's inputs that may transform into a database attack. In this technique a filter program redirects the user's input to the proxy server before it is sent to the application server. At the proxy server, data cleaning algorithm is triggered using a sanitizing application.

6 Scope

The policy-based proxy agent must be able to detect any attempt by the user to perform SQLi or XSS and either terminate the request or sanitize the input.

It must be able to at least avoid basic SQLi attacks such as Authentication bypass and Error based SQLi, and stored XSS attacks.

7 Algorithmic Steps

1) The client makes a request to a web server.

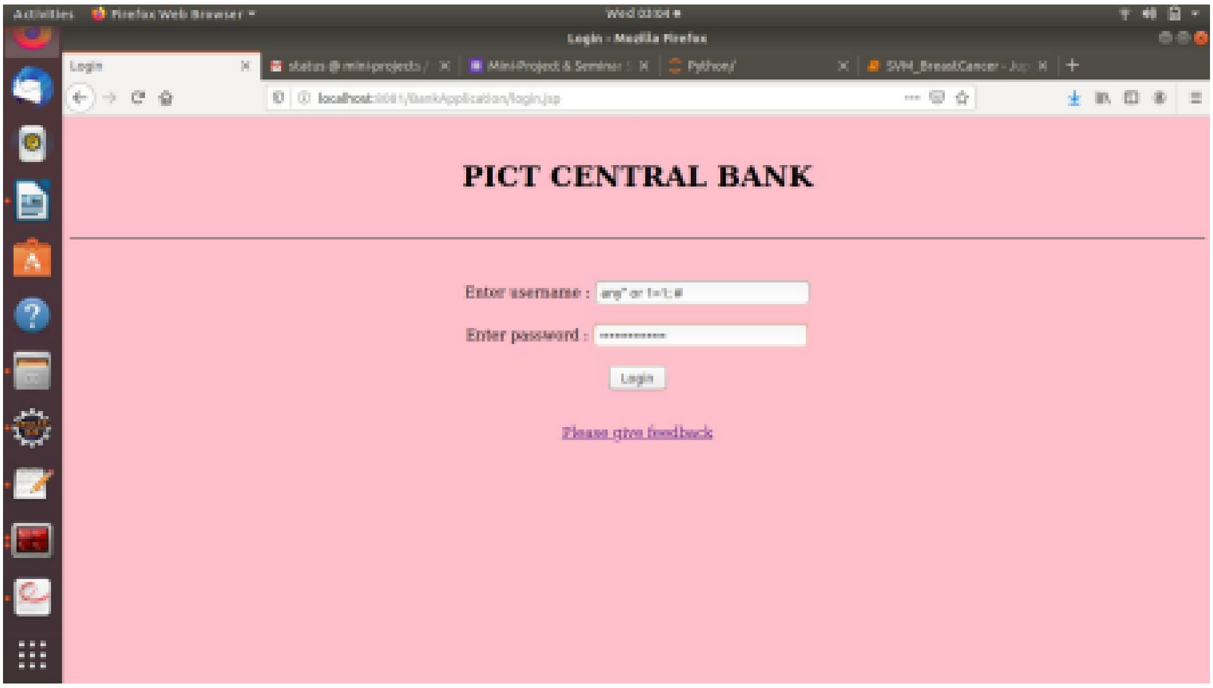
2) The web server before sending the request to the appropriate process forwards that request to a proxy server.

3) This proxy server detects whether the request contains any data which may lead to an SQLi attack or XSS attack. Thereby, classifying the request as either scripted request or query based request.

4) If the request does not contain any harmful data it is sent to the appropriate process which processes the request and sends the response back to the client.

5) If the request is found to contain harmful data, it is redirected to the client.

8 Screenshots



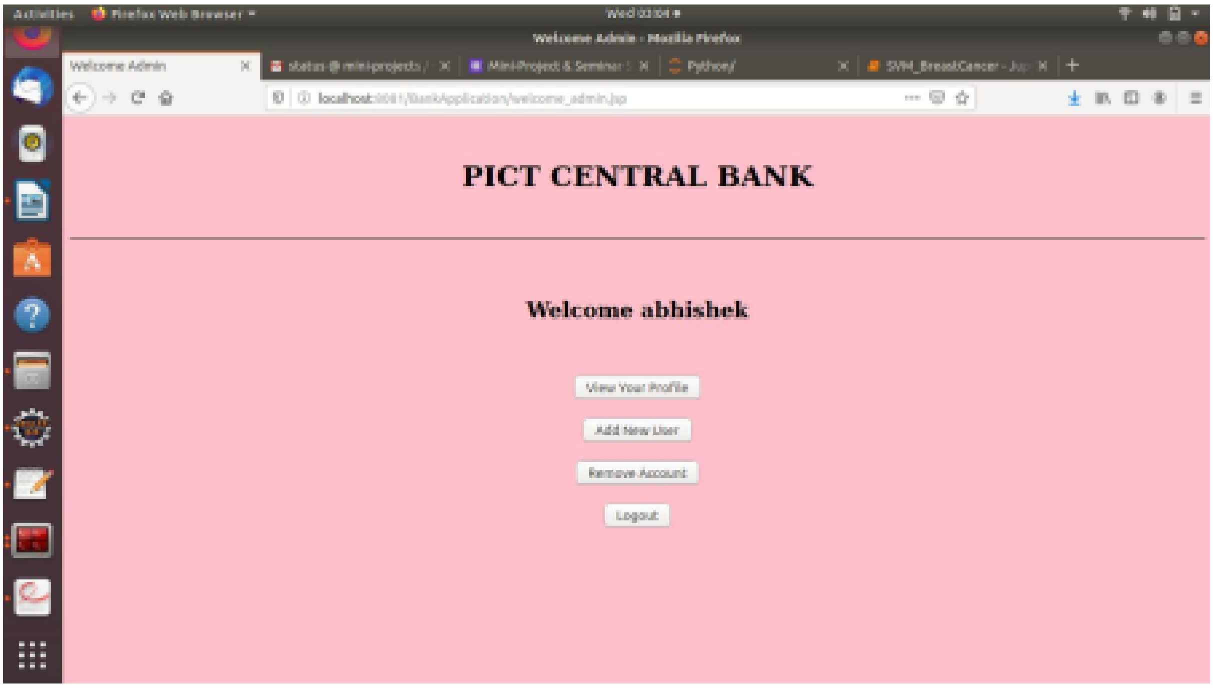
PICT CENTRAL BANK

Enter username : any' or 1=1;#

Enter password : •••••••••••

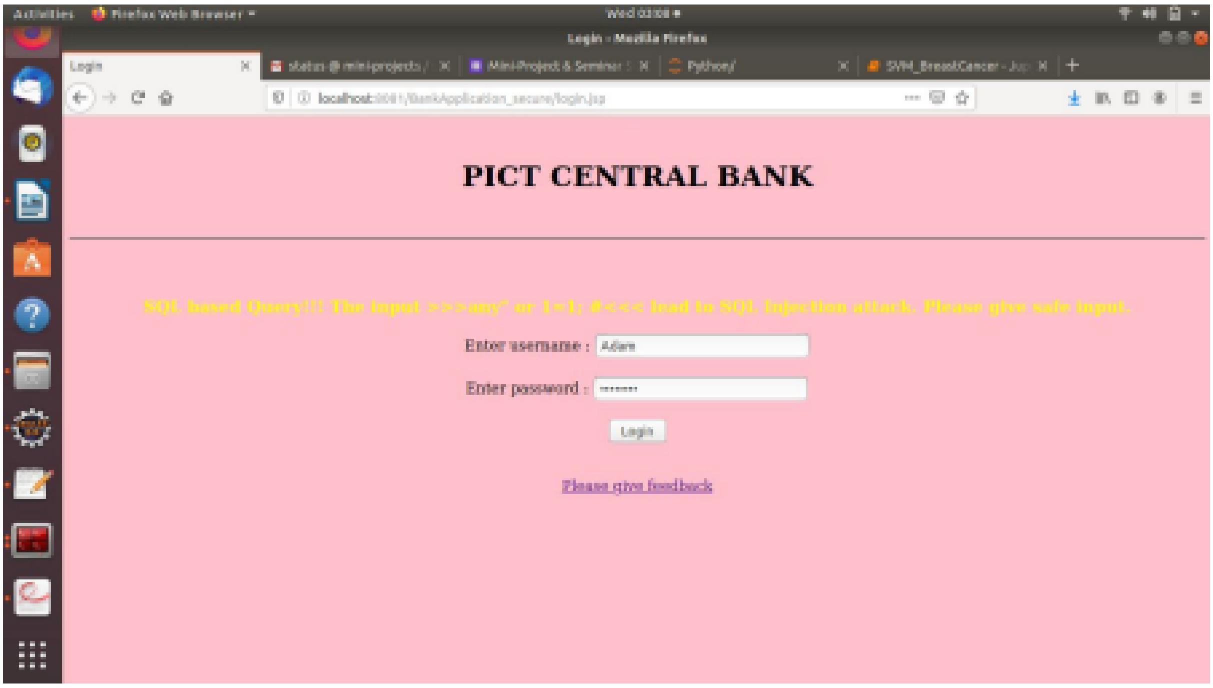
Login

Please give feedback



PICT CENTRAL BANK

Welcome abhishek



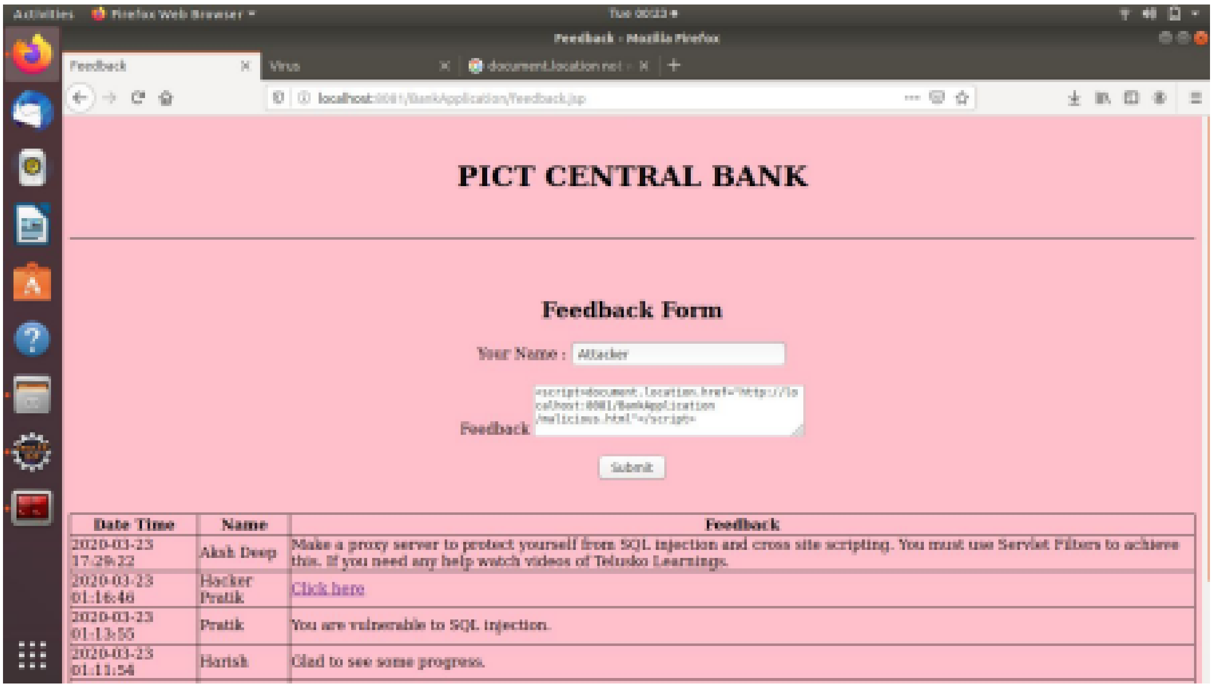
PICT CENTRAL BANK

SQL based Query!!! The input >>>any' or 1=1; #<<< lead to SQL Injection attack. Please give safe input.

Enter username : Adam

Enter password : ••••••••

Please give feedback



PICT CENTRAL BANK

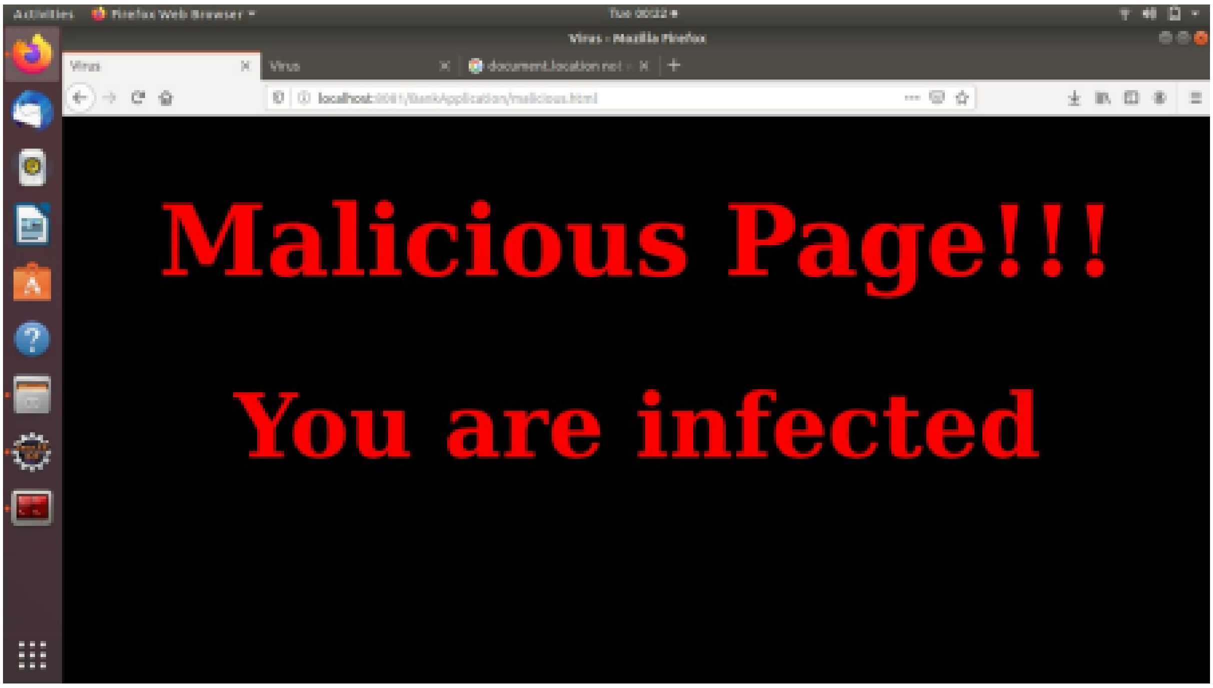
Feedback Form

Your Name : Attacker

Feedback
```
<script>document.location.href="http://lo
calhost:8081/BankApplication
/malicious.html"</script>
```

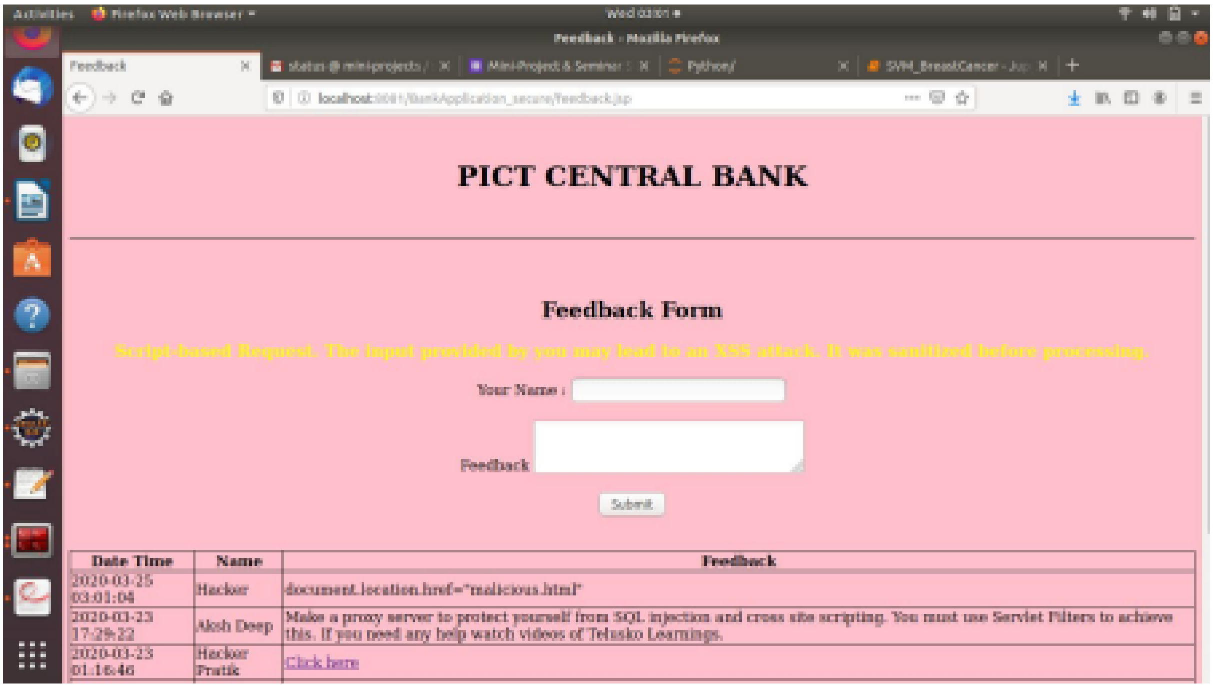
Submit

Date Time	Name	Feedback
2020-03-23 17:29:22	Aksh Deep	Make a proxy server to protect yourself from SQL injection and cross site scripting. You must use Servlet Filters to achieve this. If you need any help watch videos of Telusko Learnings.
2020-03-23 01:16:46	Hacker Pratik	Click here
2020-03-23 01:13:55	Pratik	You are vulnerable to SQL injection.
2020-03-23 01:11:54	Harish	Glad to see some progress.



Malicious Page!!!

You are infected



PICT CENTRAL BANK

Feedback Form

Script-based Request. The input provided by you may lead to an XSS attack. It was sanitized before processing.

Your Name :

Feedback

Submit

Date Time	Name	Feedback
2020-03-25 03:01:04	Hacker	document.location.href="malicious.html"
2020-03-23 17:29:22	Aksh Deep	Make a proxy server to protect yourself from SQL injection and cross site scripting. You must use Servlet Filters to achieve this. If you need any help watch videos of Telusko Learnings.
2020-03-23 01:16:46	Hacker Pratik	Click here

9 Conclusion

A policy based proxy agent was successfully developed to classify requests from a client as either scripted request or query based request and hence, protect the web application from SQLi attack and XSS attack.