

Aim – Implementation and analysis of Cross site scripting, and injection attacks.

Video Demonstration - https://youtu.be/3nCr1Y_Uya8

Things you will need

1. DVWA (Damn Vulnerable Web Application). Setup guide - <https://youtu.be/1t3gGgiDbJ8>
2. Kali Linux.

Procedure.

1. Start apache and mysql service.
2. Go to 127.0.0.1/DVWA-master/login.php
3. Login with credentials id-admin, pass-password.
4. Set DVWA security to LOW level.

Attacks:

1. SQL INJECTION ON DVWA LOW LEVEL

SQL injection is a web security vulnerability that allows an attacker to interfere with the queries that an application makes to its database. It generally allows an attacker to view data that they are not normally able to retrieve. This might include data belonging to other users, or any other data that the application itself is able to access. In many cases, an attacker can modify or delete this data, causing persistent changes to the application's content or behavior.

Attack scenario

A screenshot of a web form for user login. It features a label 'User ID:' followed by a text input field. To the right of the input field is a button labeled 'Submit'.

User id contains 5 user Ids and their information.

1' and 1=1# which gives first information of id and 1=1 gives that the query is true.

1' and 1=1 union select null,table_name from information_schema.tables#

which gives information of first id and show all table name from the database

2. REFLECTED XSS ON DVWA LOW LEVEL

Cross-site scripting (also known as XSS) is a web security vulnerability that allows an attacker to compromise the interactions that users have with a vulnerable application. It allows an attacker to circumvent the same origin policy, which is designed to segregate different websites from each other. Cross-site scripting vulnerabilities normally allow an attacker to masquerade as a victim user, to carry out any actions that the user is able to perform, and to access any of the user's data. If the victim user has privileged access within the application, then the attacker might be able to gain full control over all of the application's functionality and data.

Attack scenario

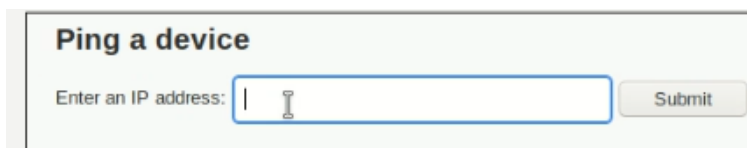
A screenshot of a web form. It has a light green background. The text "What's your name?" is on the left. To its right is a white text input field with a thin grey border. To the right of the input field is a grey button with the word "Submit" in white text.

`<script>alert(document.domain)</script>` this script will show popup with domain name

3. COMMAND INJECTION ON DVWA LOW LEVEL

OS command injection (also known as shell injection) is a web security vulnerability that allows an attacker to execute arbitrary operating system (OS) commands on the server that is running an application, and typically fully compromise the application and all its data. Very often, an attacker can leverage an OS command injection vulnerability to compromise other parts of the hosting infrastructure, exploiting trust relationships to pivot the attack to other systems within the organization.

Attack Scenario

A screenshot of a web form titled "Ping a device". Below the title, it says "Enter an IP address:". To the right of this text is a white text input field with a thin grey border. To the right of the input field is a grey button with the word "Submit" in white text.

Underlying code does not check if \$target matches an IP Address.

No filtering on special characters.

in Unix/Linux allows for commands to be separated.

payload - 127.0.0.1 | ls

this will list the files in current directory



Blog - <http://phcet.rohanparab.com/BEIT/2>

Video - https://youtu.be/3nCr1Y_Uya8

Github- <https://github.com/anonymous9396/Attacks-on-DVWA-Low-Level->