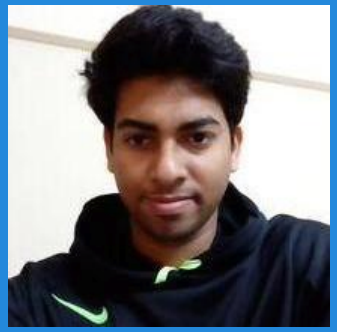
XSS: Cross Site Scripting

OpenSecurity

Ajin Abraham
Security Enthusiast | opensecurity.in

WHO AM I



- Security Researcher
- Application Security Engineer
- Founder of OWASP Xenotix XSS Exploit Framework
- Blogs at **opensecurity.in**

Agenda

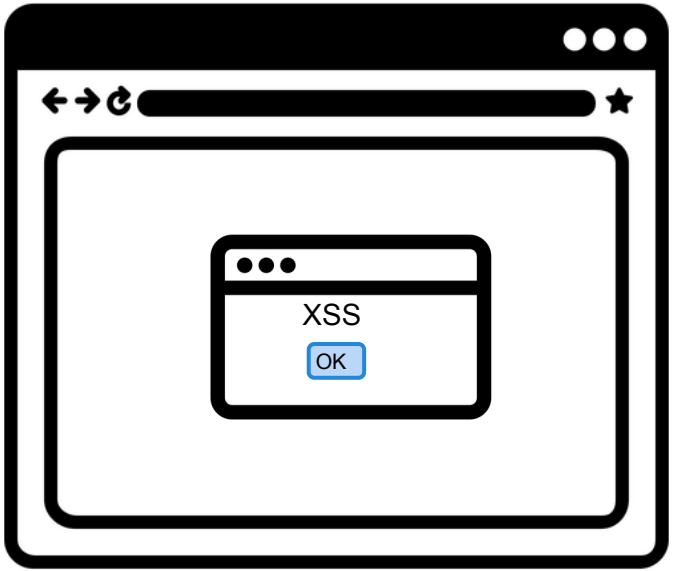
- What is XSS?
- Why XSS?
- Types of XSS
  - Reflected XSS or Non-Persistent XSS
  - Stored XSS or Persistent XSS
  - DOM XSS
    - mXSS or Mutation XSS
  - RPO or Relative Path Overwrite XSS
- What are the Source of XSS?
- Different Contexts in XSS
  - HTML Context
  - Attribute Context
  - URL Context
  - Style Context
  - Script Context
- Attacks in Real World
- Exploiting XSS with OWASP Xenotix XSS Exploit Framework
- XSS Protection

What?

XSS or **Cross Site Scripting** is a web application vulnerability that occurs when untrusted data from the user is processed by the web application without validation and is reflected back to the browser without encoding or escaping, resulting in code execution at the browser engine.

Why XSS

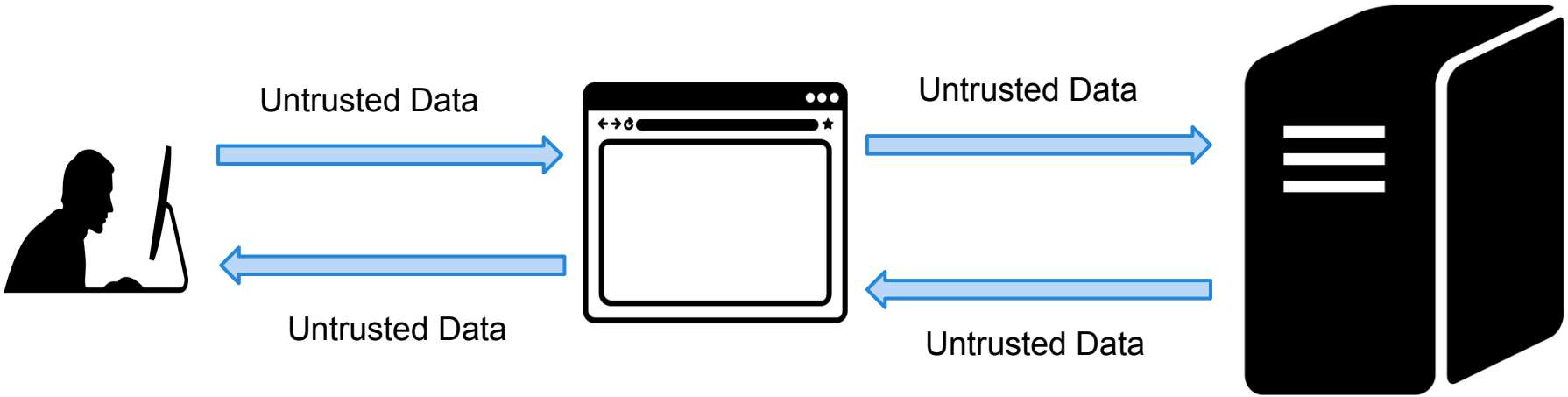
Because Code Executes in Browser and Browser cannot determine Code and Data apart



Types of XSS

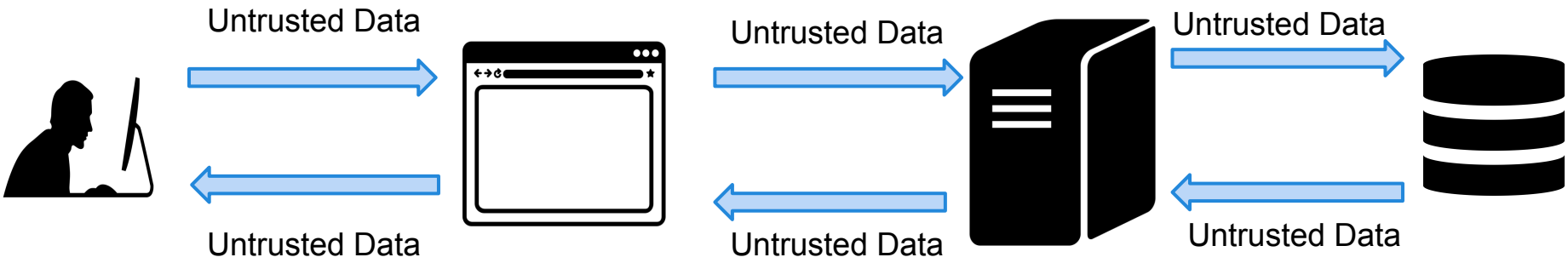
- Reflected XSS or Non Persistent XSS
- Stored XSS or Persistent XSS
- DOM XSS
  - mXSS or Mutation XSS
- RPO XSS or Relative Path Overwrite XSS

Reflected XSS



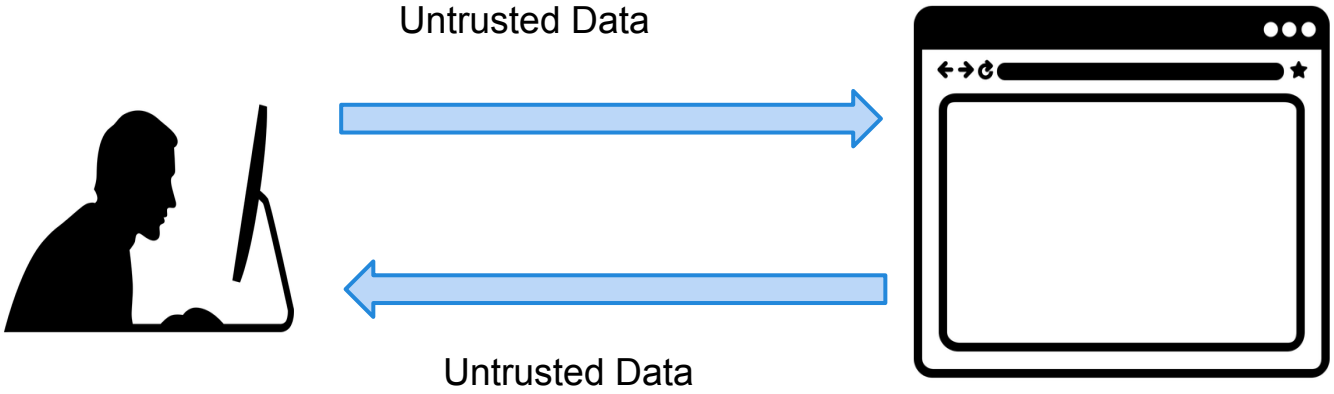
DEMO

Stored XSS


Untrusted Data
Untrusted Data
Untrusted Data
Untrusted Data
Untrusted Data
Untrusted Data

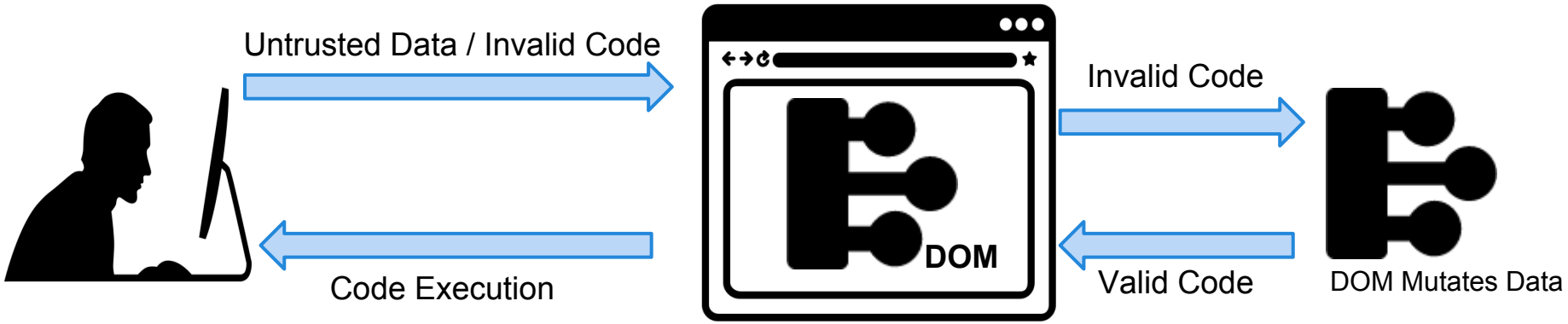
DEMO

DOM XSS



DEMO

mXSS or Mutation XSS



Works in older versions of IE

DEMO

- XSS vector that is mutated from a safe state into an unsafe unfiltered state.
- The most common form of mXSS is from incorrect reads of innerHTML.
- To Learn More: http://www.thespanner.co.uk/2014/05/06/mxss/

```
<div id=x>&lt;img src=z onerror=alert(1)&gt;</div>
<script>alert(document.getElementById('x').innerHTML)</script>
```

- DEMO: http://www.businessinfo.co.uk/labs/mxss/

RPO or Relative Path overwrite XSS

Depends on three things.

1. Stored XSS that allows CSS Injection.
2. URL Rewriting (DEMO: http://www.webdevelopersnotes.com/graphics/index.php3)
3. Relative Addressing to CSS Style Sheet

How it works

- It take advantage of CSS parser that it avoids HTML.
- URL rewriting to load CSS from the same page.
- Works in older versions of IE.
- More INFO: http://www.thespanner.co.uk/2014/03/21/rpo/
- DEMO: http://challenge.hackvertor.co.uk/xss_horror_show/chapter7/rpo.php

What are the Sources of XSS

- URL, Parameters in URL, Headers
- Form Data [From Input Boxes, Text Area etc]
- Files and Metadata (ALT, Description etc)

Different Contexts in XSS

- HTML
- Attribute
- URL Context
- Style
- Script

HTML Context

```
1   <p>
2   Hello {{Untrusted Data}}
3   </p>
4
```

```
1   <p>
2   Hello <script>alert("XSS")</script>
3   </p>
```

Exercise

Attribute Context

```
1   <img src ="image.png" alt="{{Untruted Data}}">
2   <input type ="text" value="{{Untrusted Data}}">
3   <body onload="{{Untusted Data}}">
```

```
1   <img src ="image.png" alt=" " onload=alert("xss") x=" ">
2   <input type ="text" value=" " onfocus=alert("xss") autofocus x=" ">
3   <body onload=" javascript:alert(1) ">
```

Exercise

URL Context

```
<iframe src="{{Untrusted Data}}">
<a href="{{Untrusted Data}}">Link</a>
<META http-equiv="refresh" content="5;URL={{Untrusted Data}}">
```

```
<a href=" data:text/html;charset=utf-8;base64,PHNjcmlwdD5hbGVydCgiWFNTIik8L3NjcmlwdD4= ">Link</a>
<META http-equiv="refresh" content="5;URL= data:text/html;charset=utf-8;base64,PHNjcmlwdD5hbGVydCgiWFNTIik8L3NjcmlwdD4= ">
<iframe src=" javascript:alert(1) ">

Base64 (<script>alert("XSS")</script>) = PHNjcmlwdD5hbGVydCgiWFNTIik8L3NjcmlwdD4=
```

Exercise

Style Context

```
1  <img style="{{Untrusted Data}}" src="image.png">
2  <style>
3  {{Untrusted Data}}
4  </style>
```

```
1  <img style=" xss:expression(alert('XSS')) " src="image.png">
2  <style>
3   body{width:expression(alert(1))}
4  </style>
5
```

Exercise

Script Context

```
1  <body onload="javascript:{{Untrusted Data}}">
2  <script>
3  var x='{{Untrusted Data}}';
4  </script>
```

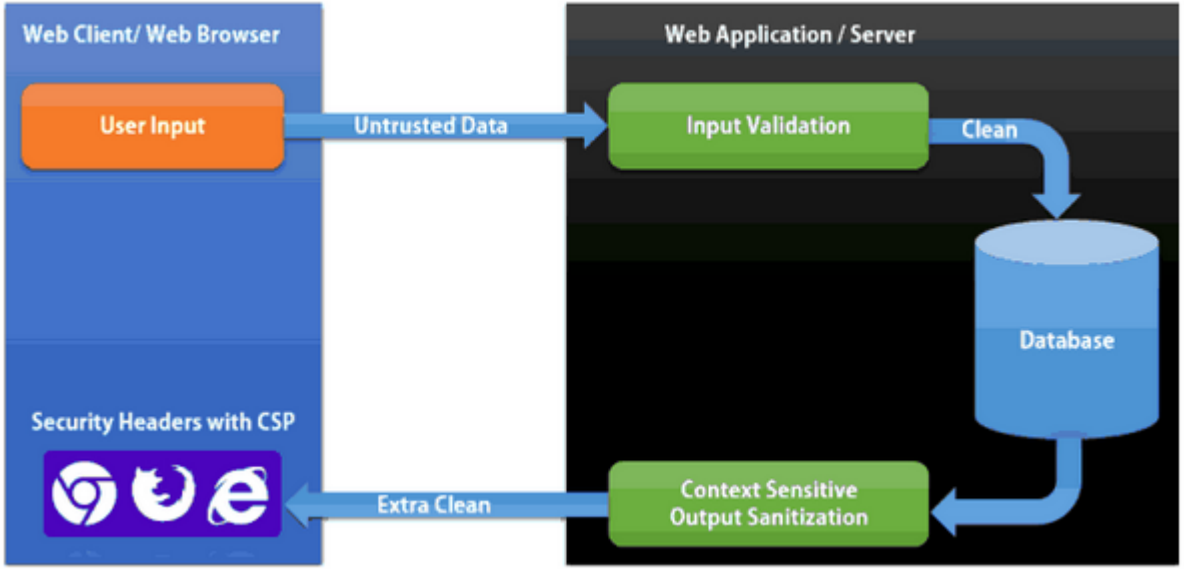
```
1  <body onload="javascript: prompt('XSS') ">
2  <script>
3  var x=' '; alert(1);// ';
4  </script>
```

Exercise

Attacks : Real World

DEMO with
OWASP Xenotix XSS Exploit Framework

XSS Protection



XSS Protection Cheatsheet

OWASP https://www.owasp.org/index.php/XSS_%28Cross_Site_Scripting%29_Prevention_Cheat_Sheet

OpenSecurity

http://opensecurity.in/the-ultimate-xss-protection-cheat-sheet-for-developers/

Thanks

“There are **10** types of people in this world. Those who understand binary and those who don't”