University of BRISTOL

Computer System- B Security

Introduction to Web Security

Part 1

Alma Oracevic

bristol.ac.uk

We will learn about…

We will learn about...

- Basics of Web application and deployment

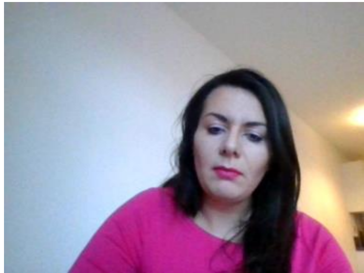
We will learn about...

- Basics of Web application and deployment
- Web vulnerabilities
  - SQL injection
  - XSS
  - CSRF
  - …

Background

- HTTP – de facto protocol when talking about WEB.
- Historically designed for static contents.
- Security was never a concern.
- Based on a *simple* client-server model.
- This is not what we see in today’s Web Applications.
- Highly technical and complex in nature

Typical web model



Typical web model

- Interaction between a browser and server (+ other stuff)

Typical web model

- Interaction between a browser and server (+ other stuff)
- Modern web pages allow personalized dynamic contents.

Typical web model

- Interaction between a browser and server (+ other stuff)
- Modern web pages allow personalized dynamic contents.
- Web pages may also run client-side scripts that "change" the Internet browser into an interface.

Typical web model

- Interaction between a browser and server (+ other stuff)
- Modern web pages allow personalized dynamic contents.
- Web pages may also run client-side scripts that “change” the Internet browser into an interface.
- Modern web sites allow the capture, processing, storage and transmission of sensitive customer data.

HTTP protocol

HTTP protocol

- HTTP is a stateless protocol

HTTP protocol

- HTTP is a stateless protocol
- HTTP URL: host/dir/resource
  - Host to IP (DNS)

HTTP protocol

- HTTP is a stateless protocol
- HTTP URL: host/dir/resource
  - Host to IP (DNS)
- HTTP requests
  - GET (part of the URL)
  - POST (part of the header body)
  - Because of stateless property, a fresh request is made with no memory of the previous interactions.

HTML

- HTTP request and response are rendered in HTML
- HTML forms
  - Allow for using key-value pairs to be processed by the server
  - Together with other languages (LabaScript/PHP) provide a very powerful interaction mode
  - img, iframe, href, etc.

Static vs Dynamic pages (1)

Static (ages

- Static pages are a typical HTML + CSS assisted Rendered the same content each time
- Only way to change is to manually change the server side page!
- Interaction is via hyperlinks on the page.

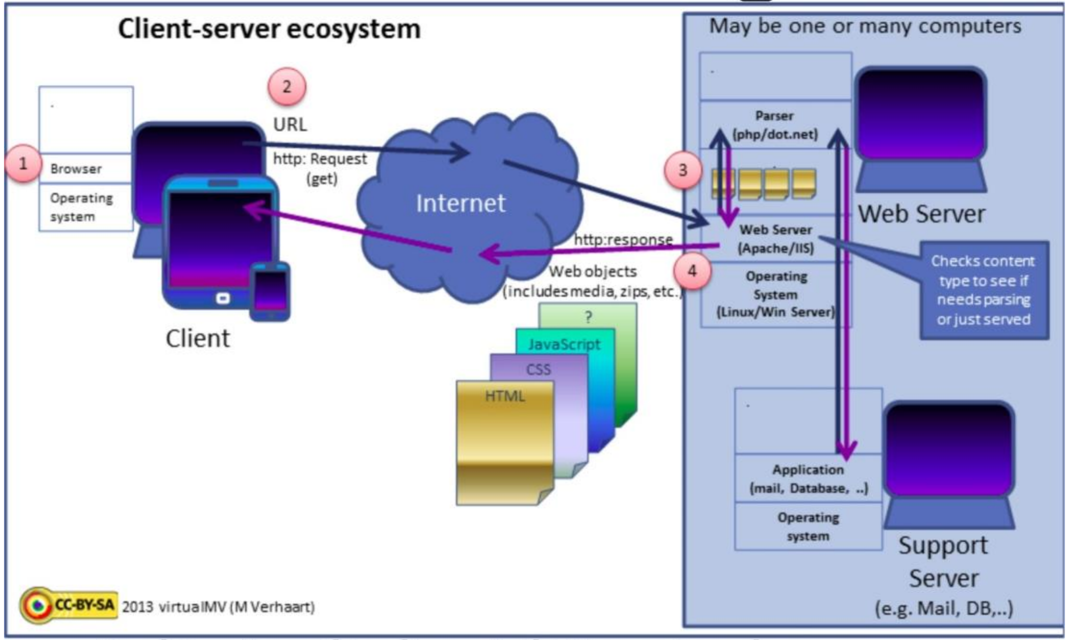
  Was good from security point of view!

Static vs Dynamic pages (2)

Dynamic pages (from https://en.wikipedia.org/wiki/Dynamic_web_page)

- Are interactive in the sense that based on the request parameters, a new page is rendered (server side).
- Pages contain other scripting code (JavaScript) that changes rendering on the client-side (client side)
- It also involves other entities, like application serves, DB etc.

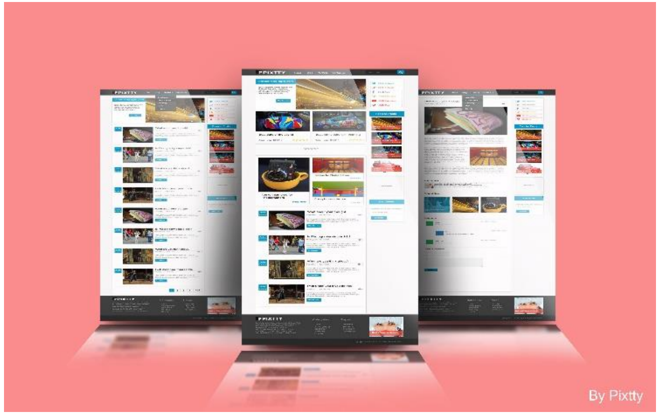
Typical Dynamic webpage rendering



Src: https://en.wikipedia.org/wiki/Dynamic_web_page

So..

- Web applications are computer programs allowing website visitors to submit and retrieve data *to/from a database, for example, over the Internet* using their preferred web browser.



By Pixtty

Java Script …

- JavaScript is one form of client side script that permits dynamic elements on each page.
- The web browser is key – it interprets and runs all scripts!!
- All requests and responses are nothing but codes written in various languages/scripts.
- And, as we have seen, codes are powerful and dangerous, if not managed!!

Some features

- HTML for look and feel
- JavaScript- a powerful language for dynamic content

```
<html>
    …
    <script> javascript code </script>


        …
    </html>
```

Some features

- JavaScript in the **Browser** is sandboxed.

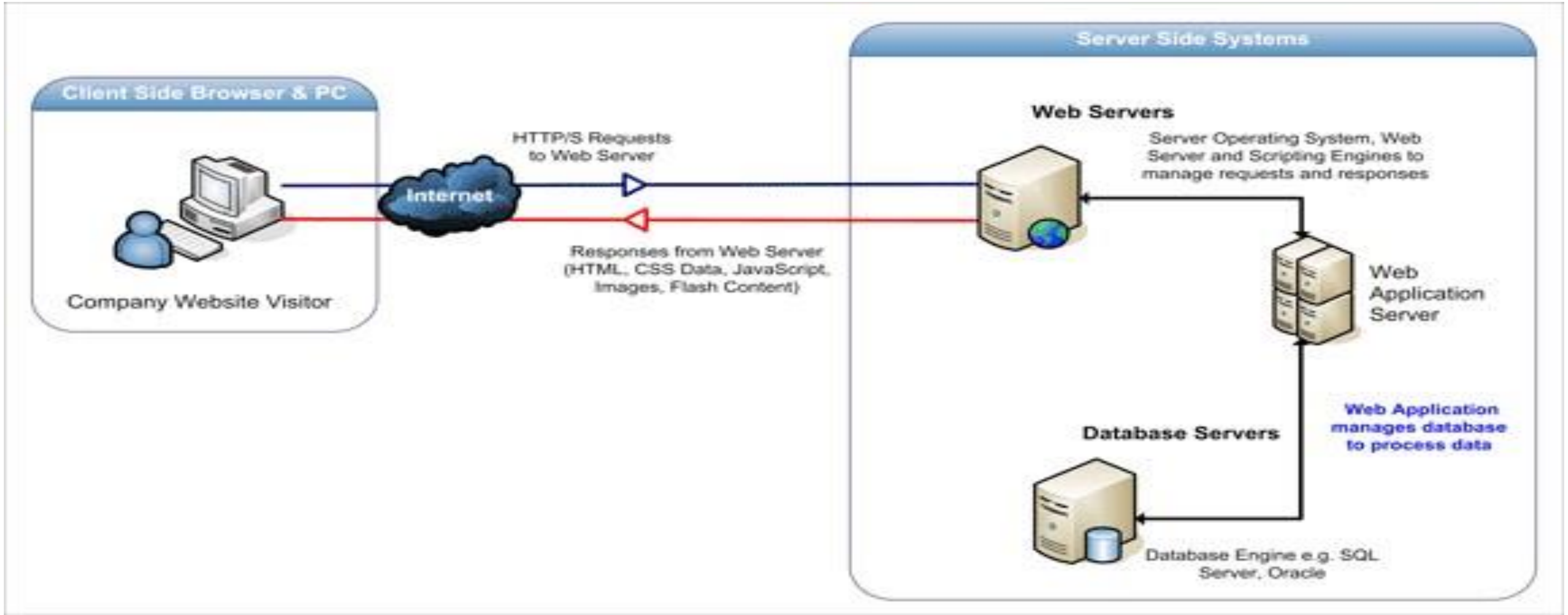
So??

- We also know that now a days, many websites stores data on local machine, e.g. cookies, user data (auto fill), passwords etc.
- JavaScript can read resources -> we can steal any information???
- There are security mechanisms to take care of it!!

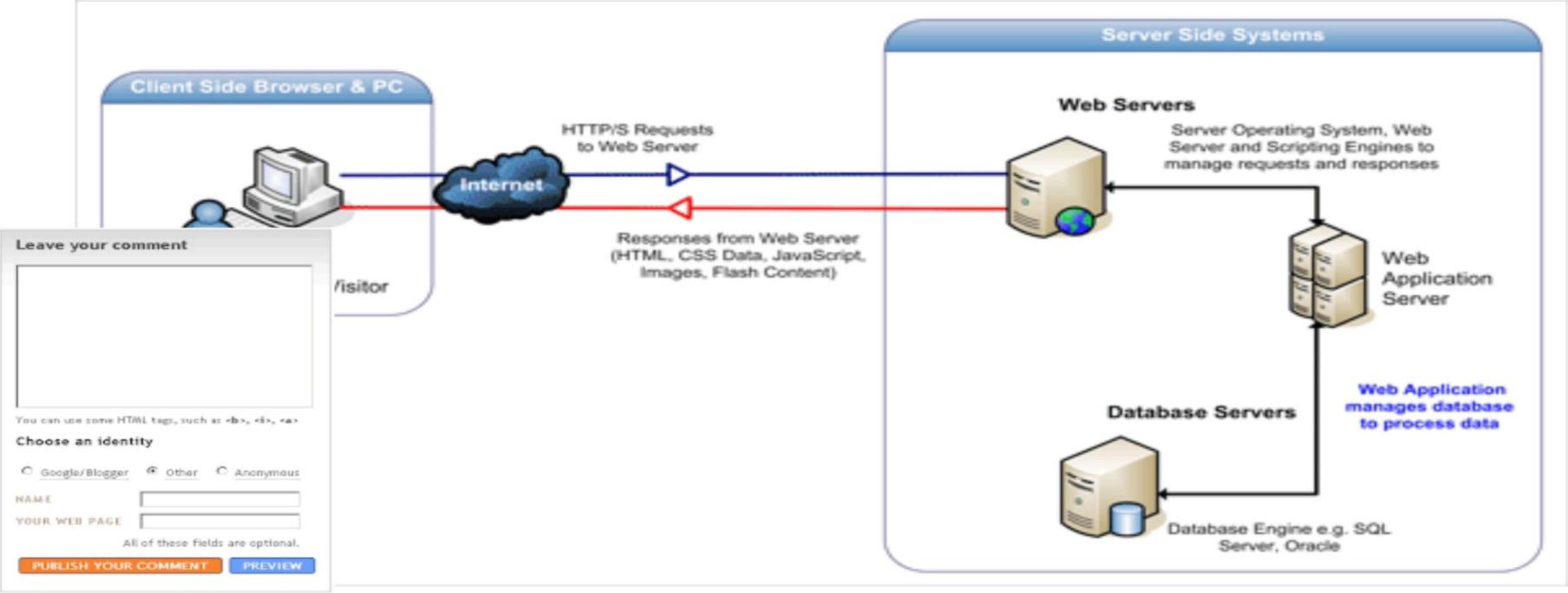
Same Origin Policy

- For absolute URIs, the origin is the triple {protocol,host port}.
- Two resources are considered to be of the same origin if and only if all these values are exactly the same.
- Example:
  - Allowed: http://www.abc.com/doc1.html & http://www.abc.com/doc2.html
  - Not allowed: http://www.abc.com:8080/doc1.html

Typical Web Application Vulnerabilities


Server Side Systems
Client Side Browser & PC
Web Servers
HTTP/S Requests to Web Server
Server Operating System, Web Server and Scripting Engines to manage requests and responses
Internet
Responses from Web Server (HTML, CSS Data, JavaScript, Images, Flash Content)
Web Application Server
Company Website Visitor
Web Application manages database to process data
Database Servers
Database Engine e.g. SQL Server, Oracle

Typical Web Application Vulnerabilities



Leave your comment

You can use some HTML tags, such as <b>, <i>, <a>

Choose an identity

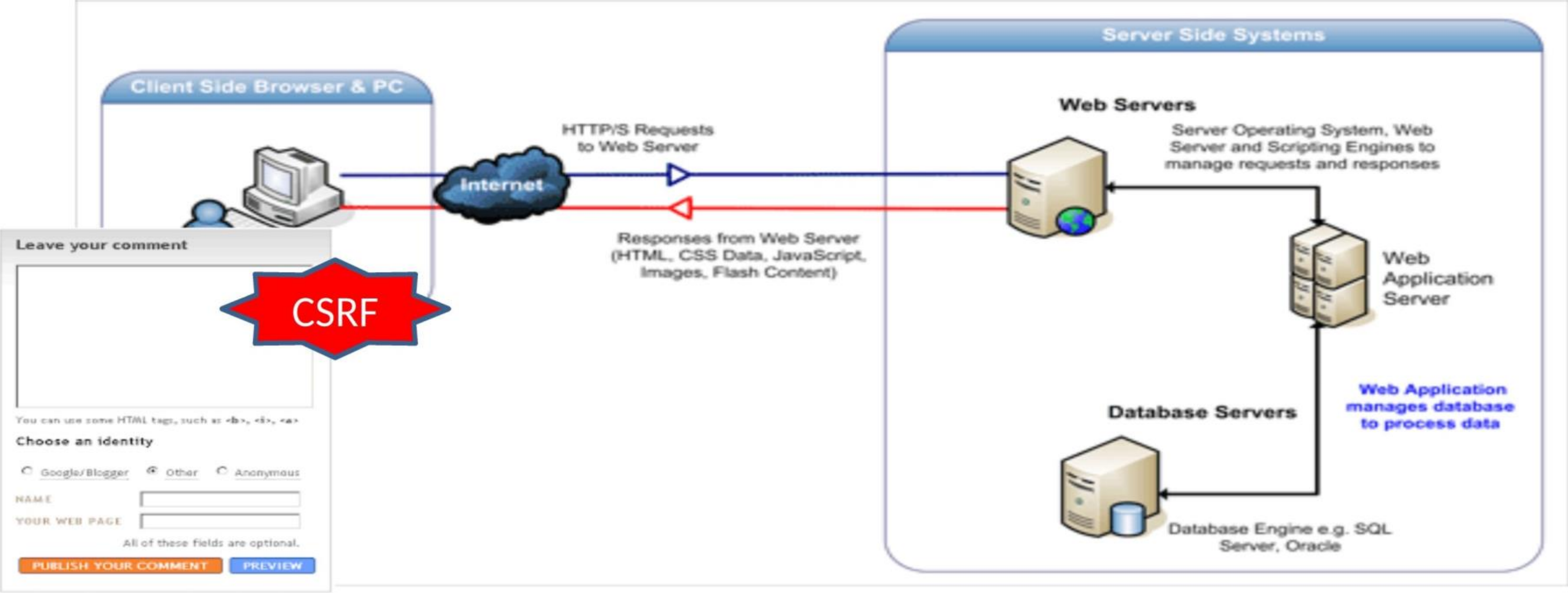
○ Google/Blogger ● Other ○ Anonymous

NAME

YOUR WEB PAGE

All of these fields are optional.

Typical Web Application Vulnerabilities


Server Side Systems
Client Side Browser & PC
Web Servers
HTTP/S Requests to Web Server
Server Operating System, Web Server and Scripting Engines to manage requests and responses
Internet
Responses from Web Server (HTML, CSS Data, JavaScript, Images, Flash Content)
Web Application Server
Web Application manages database to process data
Database Servers
Database Engine e.g. SQL Server, Oracle
CSRF

Leave your comment

You can use some HTML tags, such as <b>, <i>, <a>

Choose an identity

○ Google/Blogger ● Other ○ Anonymous

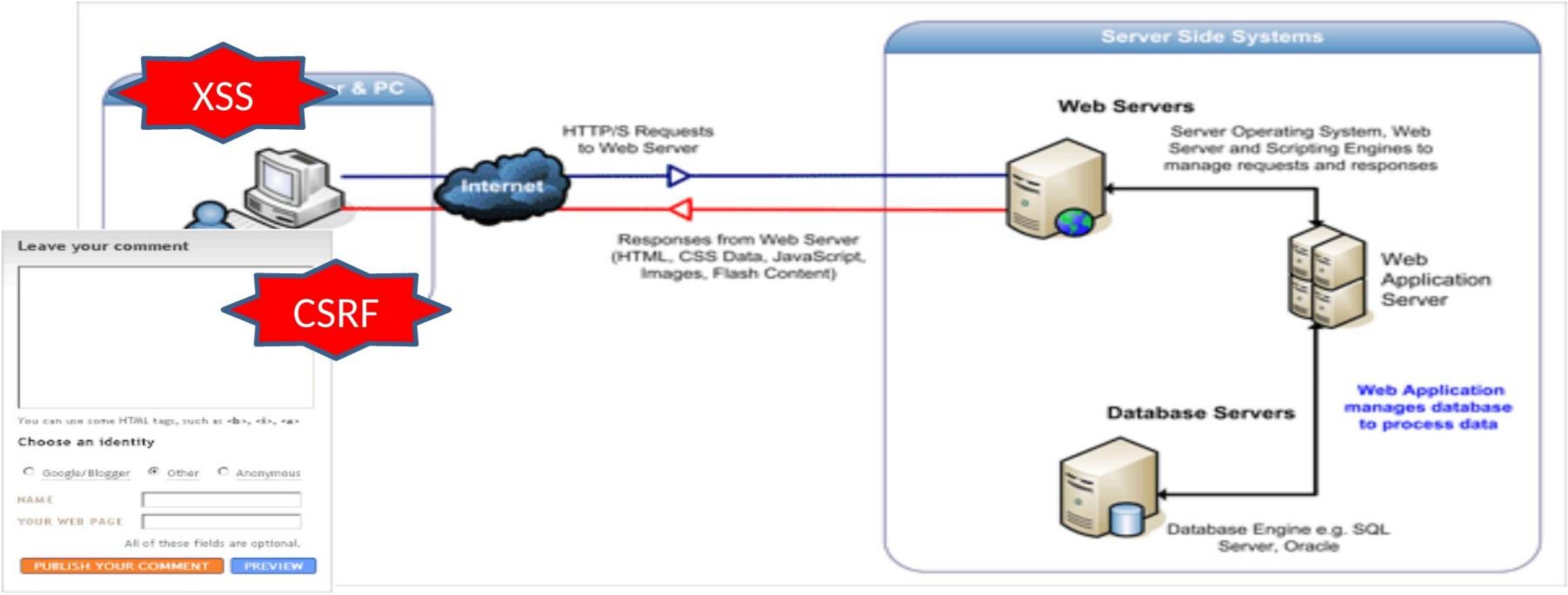
NAME

YOUR WEB PAGE

All of these fields are optional.

PUBLISH YOUR COMMENT   PREVIEW

Typical Web Application Vulnerabilities


XSS
r & PC
CSRF
Internet
HTTP/S Requests to Web Server
Responses from Web Server (HTML, CSS Data, JavaScript, Images, Flash Content)
Server Side Systems
Web Servers
Server Operating System, Web Server and Scripting Engines to manage requests and responses
Web Application Server
Web Application manages database to process data
Database Servers
Database Engine e.g. SQL Server, Oracle

Leave your comment

You can use some HTML tags, such as <b>, <i>, <a>

Choose an identity

○ Google/Blogger ● Other ○ Anonymous

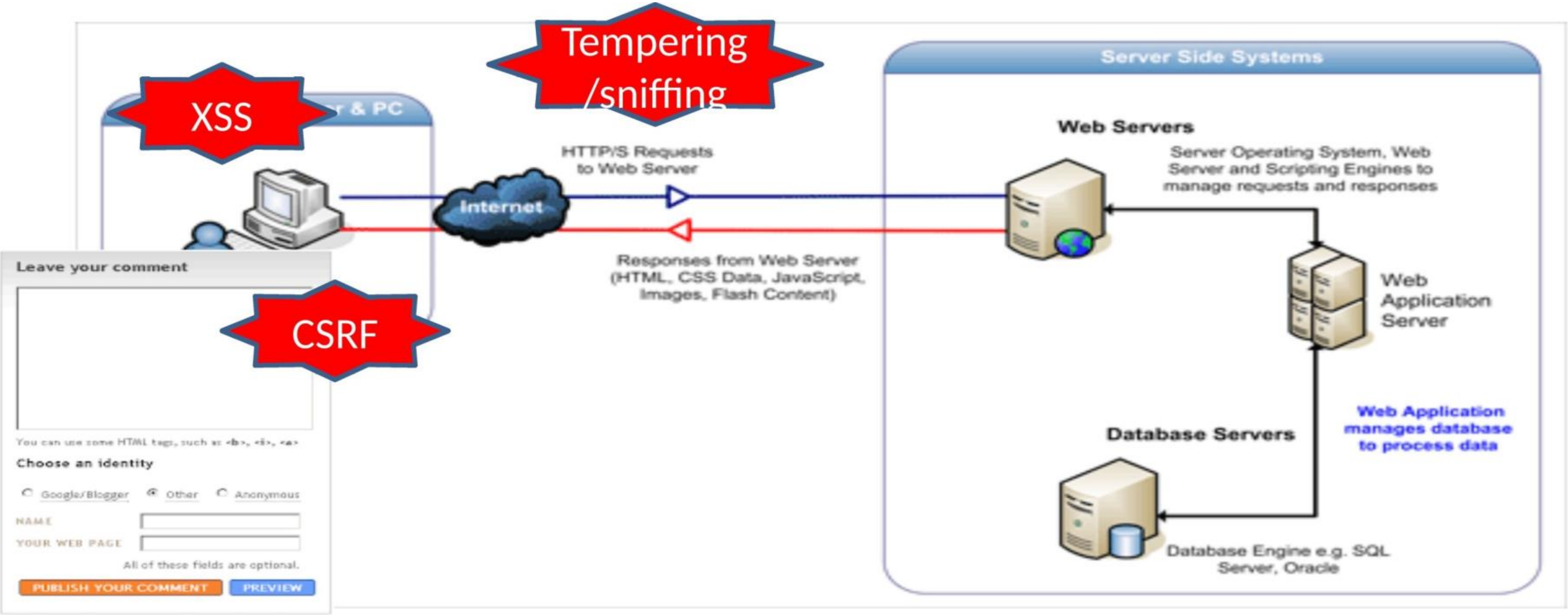
NAME

YOUR WEB PAGE

All of these fields are optional.

PUBLISH YOUR COMMENT
PREVIEW

Typical Web Application Vulnerabilities



Leave your comment

You can use some HTML tags, such as <b>, <i>, <a>

Choose an identity

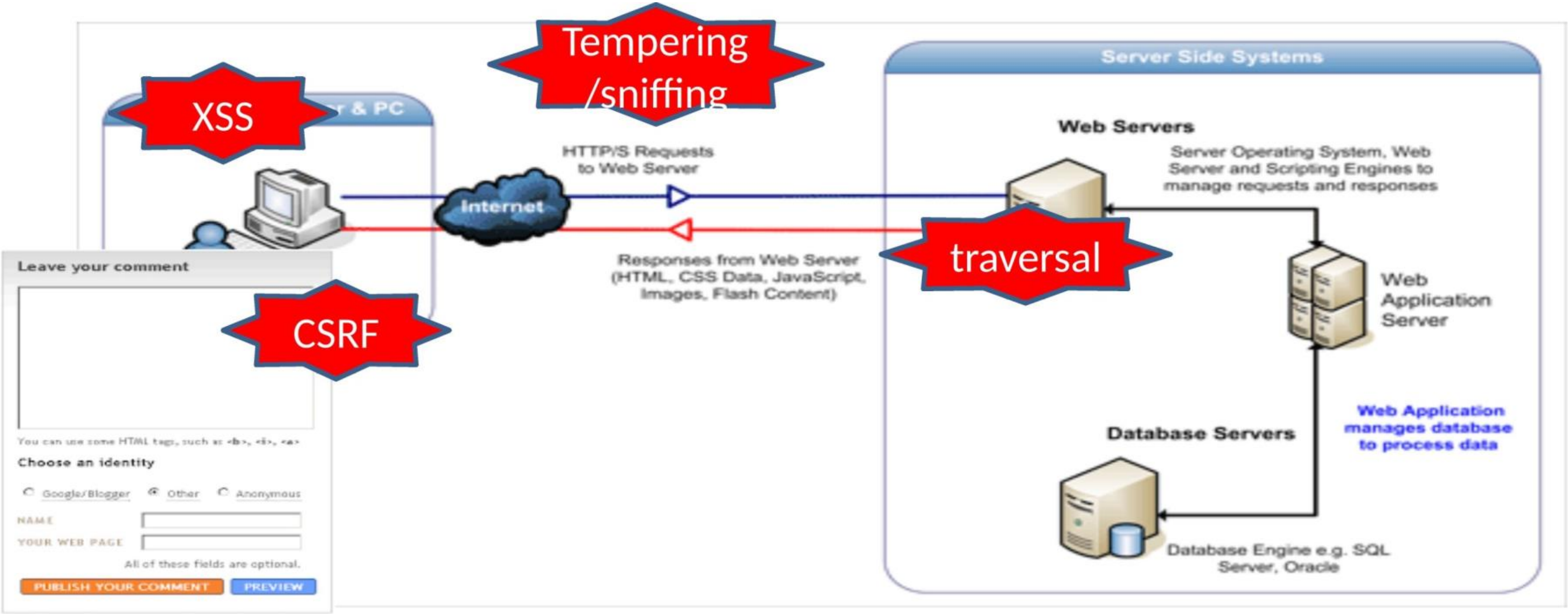
○ Google/Blogger ● Other ○ Anonymous

NAME

YOUR WEB PAGE

All of these fields are optional.

Typical Web Application Vulnerabilities



Leave your comment

You can use some HTML tags, such as <b>, <i>, <a>

Choose an identity

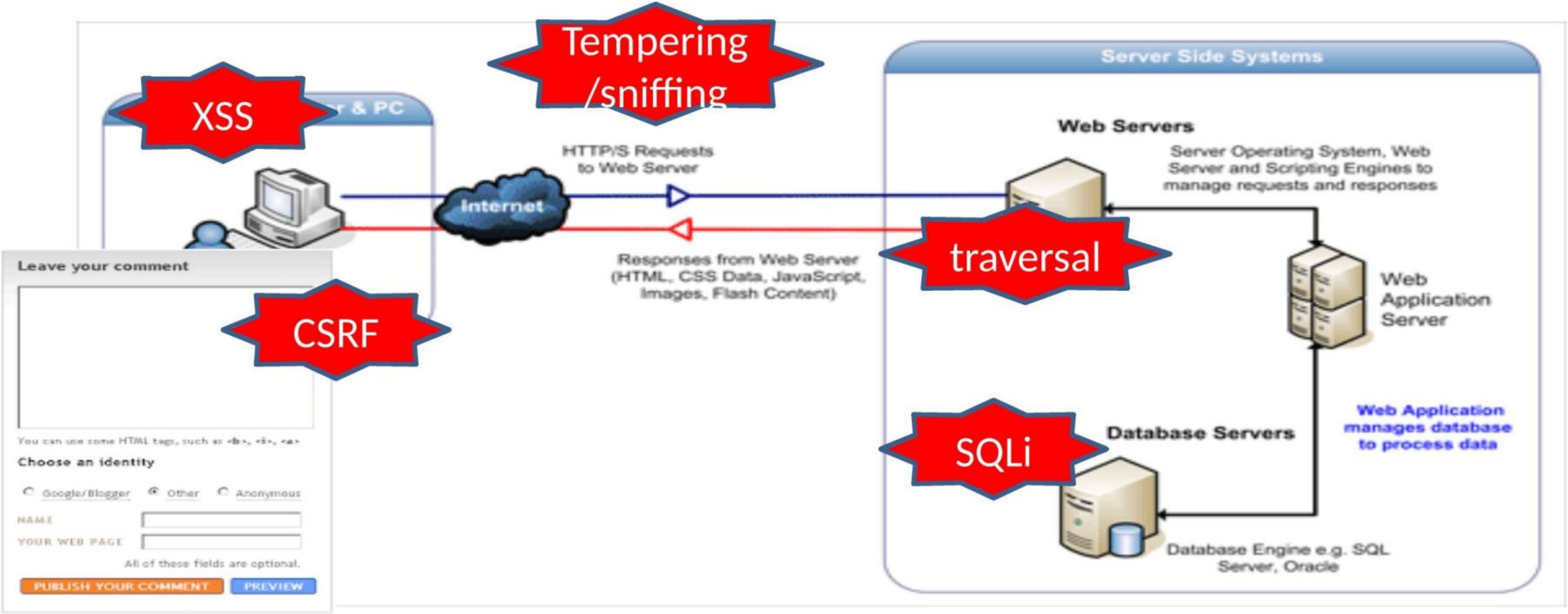
○ Google/Blogger ● Other ○ Anonymous

NAME

YOUR WEB PAGE

All of these fields are optional.

Typical Web Application Vulnerabilities



Leave your comment

You can use some HTML tags, such as <b>, <i>, <a>

Choose an identity

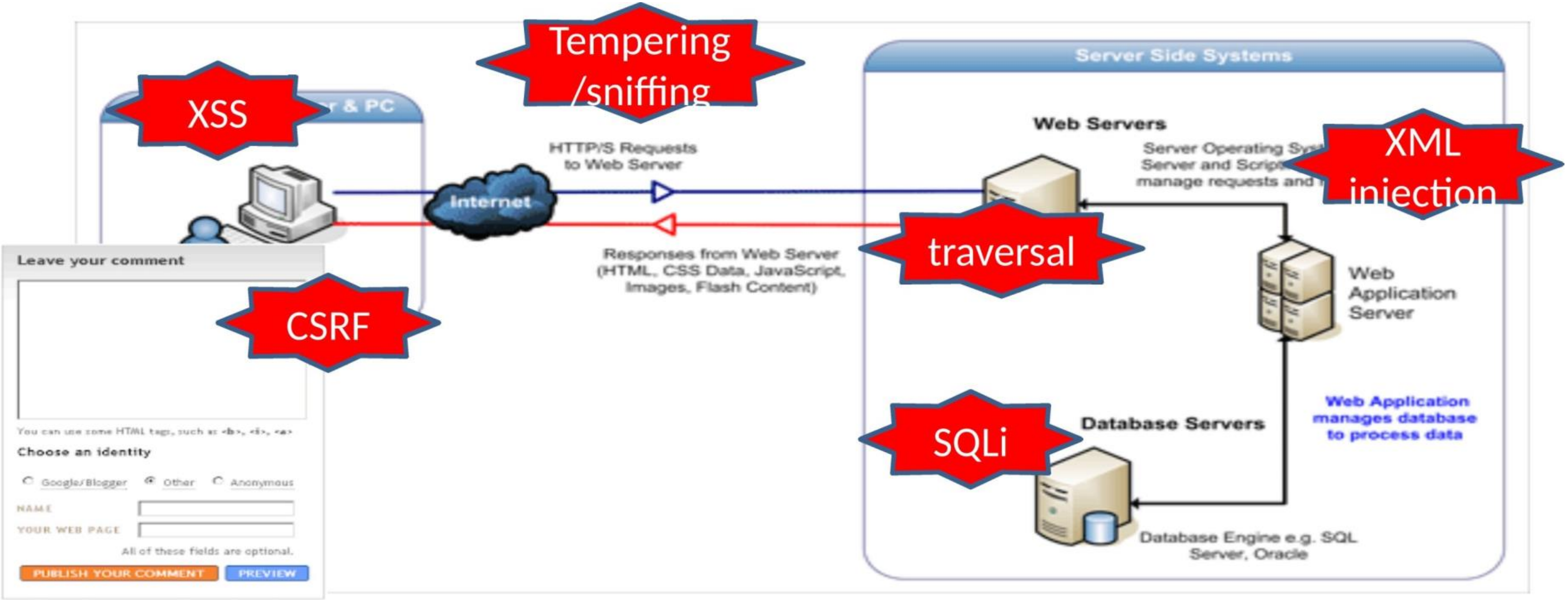
○ Google/Blogger ● Other ○ Anonymous

NAME

YOUR WEB PAGE

All of these fields are optional.

Typical Web Application Vulnerabilities



CSRF

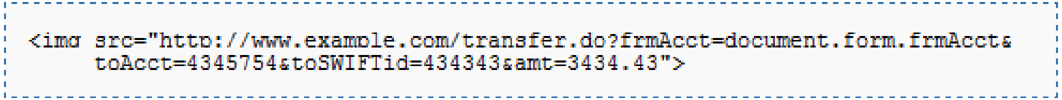
Cross Site Request Forgery (23% by IBM)

Inject code that:

- Runs in the victim’s browser
- Open a session to a vulnerable 3rd party service
  - Using the victim’s credentials

Example:

- Insert a money transfer in a page
  - Fo
  - En



```
<img src="http://www.example.com/transfer.do?frmAcct=document.form.frmAcct&
    toAcct=4345754&toSWIFTid=434343&amt=3434.43">
```

https://www.youtube.com/watch?v=m0EHlfTgGUU

CSRF made easy!

XSS

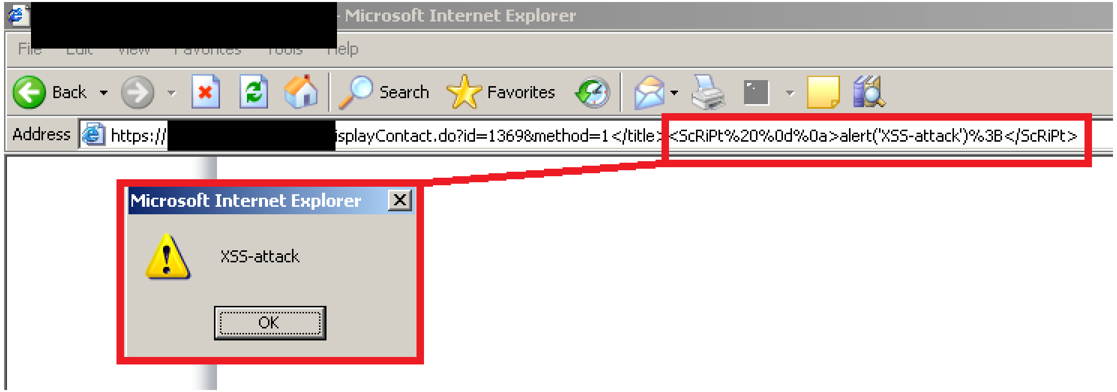
Cross Site Scripting

- Attacker can inject untrusted snippets of JS into your application without validation
- JS is then executed by the victim who visits the target site
- 3 types of XSS
  - Reflected XSS
    - Attacker sends the victim a link to the target app through email, social media, etc.
    - The link has script embedded which executes when target site is visited
  - Stored XSS
    - Attacker plants persistent script in target website…

XSS

Advanced tools are out there to abuse flaws

- Tunnel traffic using XSS
  - http://www.portcullis-security.com/uplds/whitepapers/XSSTunnelling.pdf
  - http://www.portcullis-security.com/tools/free/xssshell-xsstunnell.zip



[redacted] - Microsoft Internet Explorer

https://[redacted]isplayContact.do?id=1369&method=1</title><ScRiPt%20%0d%0a>alert('XSS-attack')%3B</ScRiPt>

Microsoft Internet Explorer

XSS-attack

XSS

Prevention

Use vetted libraries or frameworks

Use HttpOnly attribute

Input validation

Demonstration https://www.youtube.com/watch?v=i38LMZyKIqI