Web App Tool Time

Cybersecurity

15.3 Web Vulnerabilities and Hardening

Class Objectives

By the end of today's class, you will be able to:

Identify ways in which web application security tools can assist with testing security vulnerabilities.

Configure Burp Suite and Foxy Proxy to capture and analyze an HTTP request.

Identify session management vulnerabilities using the Burp Suite Repeater function.

Conduct a brute force attack against a web application login page with the Burp Intruder function.

Attacking Web Applications Using Security Tools

Last Class...

Directory Traversal

A web app back-end component vulnerability in which an attacker accesses files and directories from a web application outside a user's authorized permissions.

Local File Inclusion (LFI)

Another web app back-end component vulnerability, in which an attacker tricks the application into running unintended back-end code or scripts that are LOCAL to the application's file system.

Remote File Inclusion (RFI)

A back-end component web app vulnerability in which an attacker tricks the application into running unintended back-end code or scripts—similar to LFI, except that the scripts are REMOTE to the application's file system.

Directory traversal, LFI, and RFI all fall under the OWASP risk broken access control.

Today's Class...

Web application security tools can assist security professionals by automating testing processes.

When we first executed SQL injection attacks, we were successful after our first payload attempt of:

```
jsmith OR '1' = '1
```

Realistically, we would need to test many different payloads, such as:

```
jsmith" or true--
```

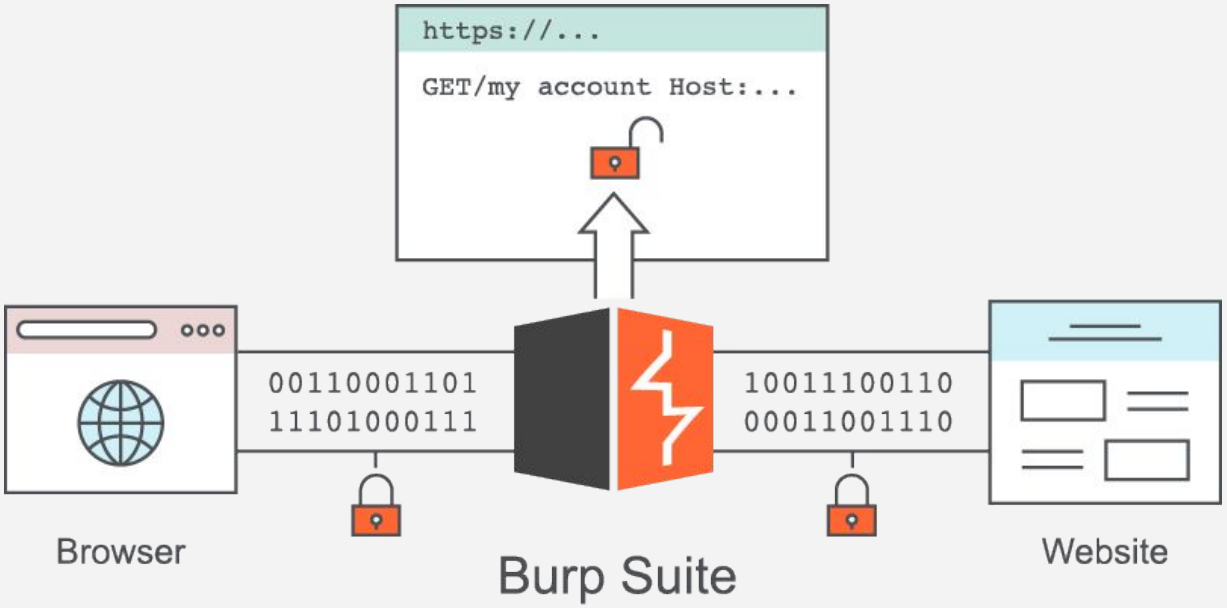
```
jsmith OR "1" = "1
```

```
jsmith -- OR '1' = '1
```

```
jsmith') or true--
```

Today's Class

Today, we will use Burp Suite, a popular web application security tool, to automate processes, capture, display, and modify web app requests and responses, and use other built-in features to test for vulnerabilities.



(portswigger.net)

Today's Class

While demonstrating Burp Suite, we will explore vulnerabilities that fall under the **OWASP risk of Identification and Authentication Failures**.

authentication methods that are used by a web application.

A01	Broken Access Control	A06	Vulnerable and Outdated Components
A02	Cryptographic Failures	A07	**Identification and Authentication Failures**
A03	Injection	A08	Software and Data Integrity Failures
A04	Insecure Design	A09	Security Logging and Monitoring Failures
A05	Security Misconfiguration	A10	Server-Side Request Forgery (SSRF)

Today's Class...

The class will proceed as follows:

01

We will begin with introducing **Burp Suite** and **Foxy Proxy** and how to configure them to capture and analyze HTTP requests and responses.

02

Then we will learn how to use the **Burp Repeater** feature to determine session hijacking vulnerabilities.

03

Lastly, we will show how to use the **Burp Intruder** feature to conduct an automated brute force attack.

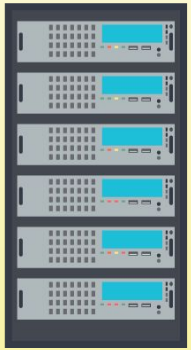
Intro to Web Proxies and Burp Suite

How Web Apps Interact with Back-End Servers

01

A user's browser (the client) requests an image of a car to be displayed with an HTTP request.

HTTP request for car image



02

The web server (the server) responds with that car image by sending an HTTP response.

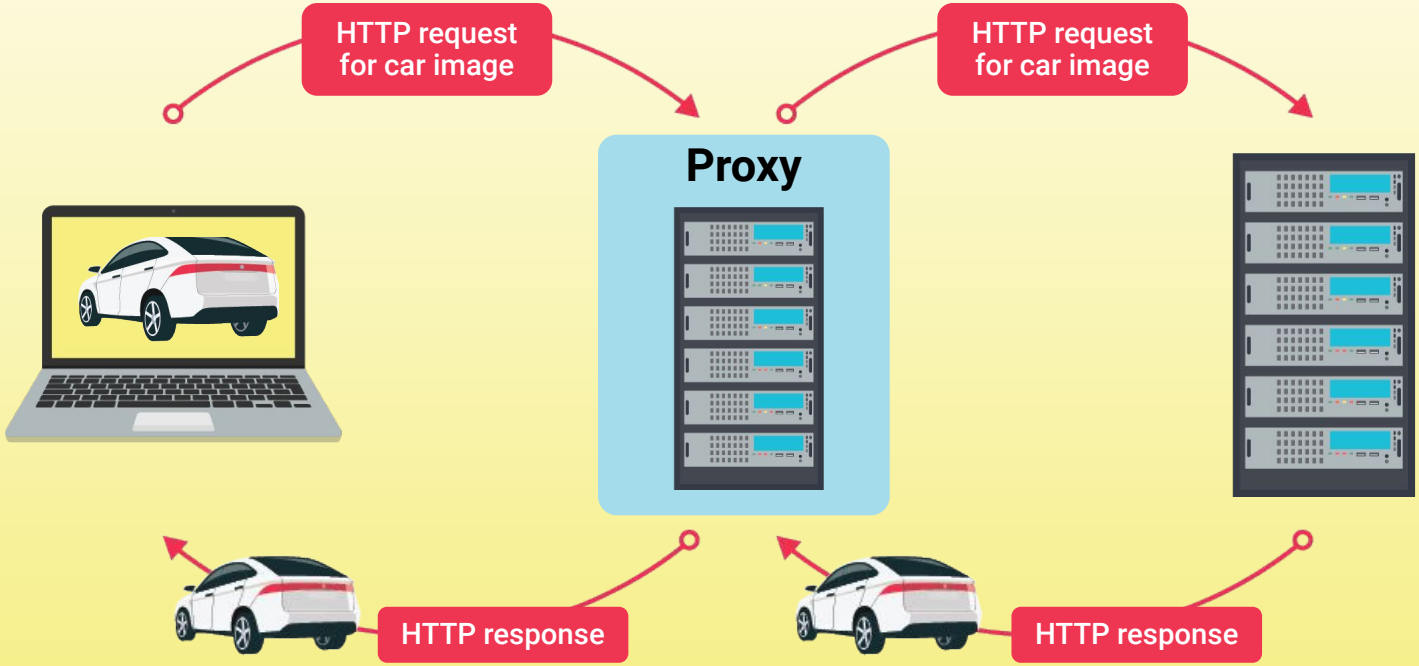
03

The browser receives the HTTP response and renders the image of the car to the user.

HTTP response

Web Proxy

A **web proxy** is an intermediary between the client and the server.
Internet traffic flows through the proxy on the way to its intended destination.

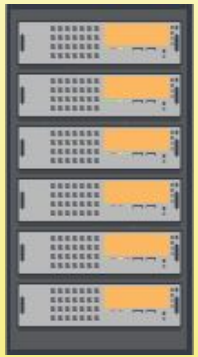


Web Proxy

Web proxies can be used by:

Organizations

To monitor and block harmful web traffic, as some web proxies can be configured to block specific websites



Individuals

To provide themselves anonymity when using the internet, as some web proxies can change the source IP address.

Burp Suite

Burp Suite is a web application security tool that lies between your browser and your target application.

- Burp Suite intercepts raw HTTP traffic either from the browser or the server, thus functioning as a web proxy.
- Burp has many additional features and capabilities to allow a security professional to analyze, modify, and automate the HTTP traffic before passing it along to its final destination.

(portswigger.net)

Burp Suite Demo

Burp Suite is a web application security tool that lies between your browser and your target application.

01 Start and access BURPSUITE

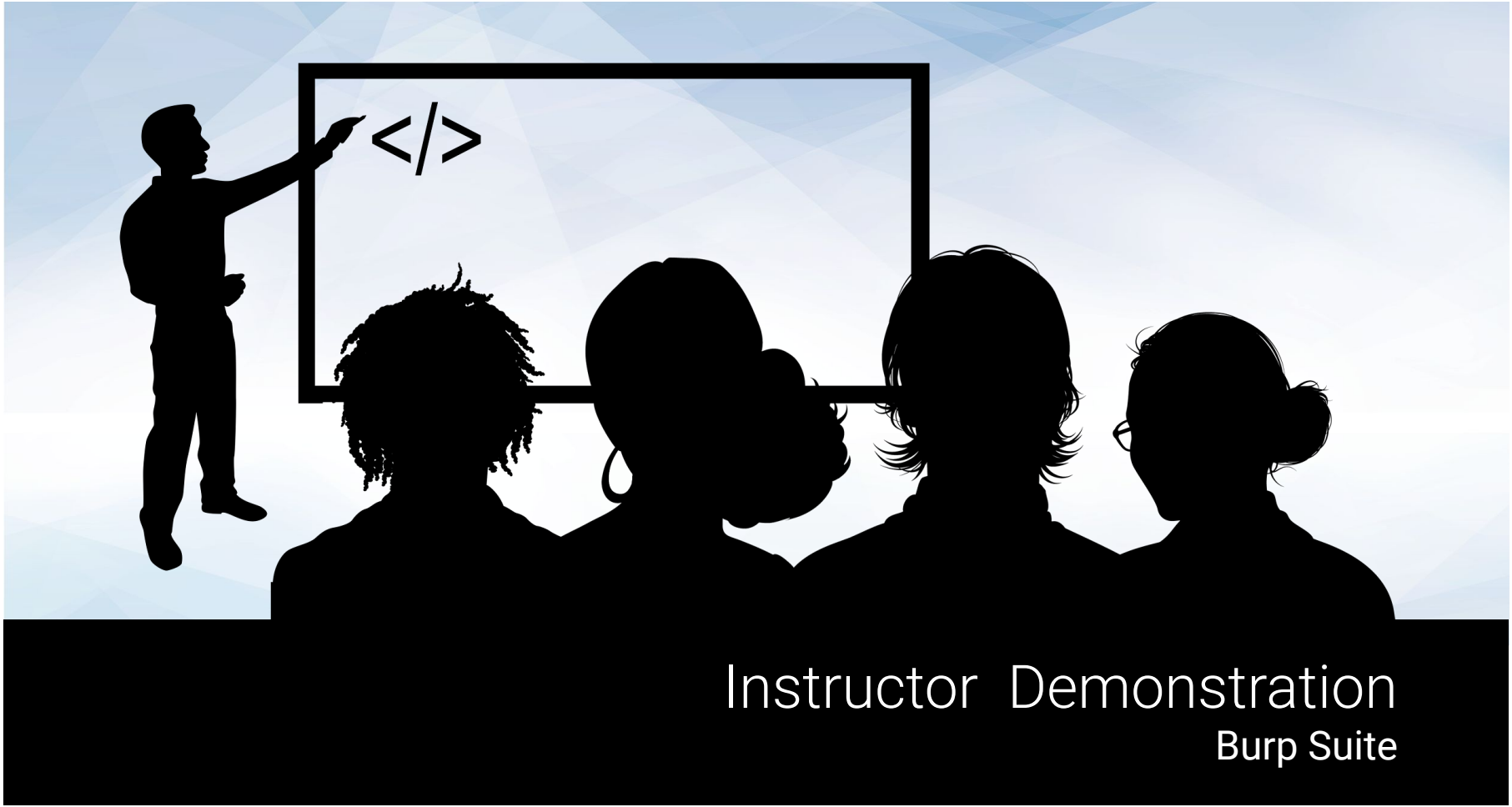
02 Navigate Burp Suite.

03 Configure the proxy on Burp Suite.

04 Configure the proxy on your browser.

05 Enable Foxy Proxy  to send traffic to Burp Suite.

06 Review the captured traffic on Burp Suite.

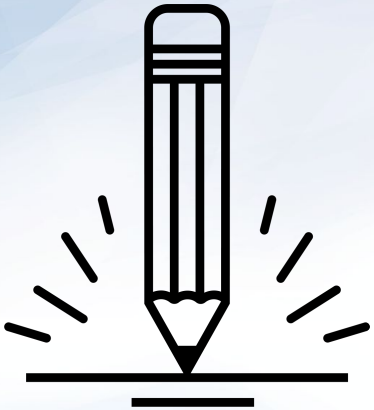


Instructor Demonstration

Burp Suite

Demo Summary

The guided tour we just completed illustrated the steps that a security professional would take to capture and display raw HTTP traffic using Foxy Proxy and Burp Suite.



Activity: Configuring Burp Suite

In this activity you'll configure Burp Suite proxy settings in order to prepare for a future session hijacking attempt.

Suggested Time:
20 Minutes

Time’s Up! Let’s Review.


Break

Session Management Vulnerabilities

Cookies and Session Management

In this section we will learn how to use a Burp Suite feature called Repeater to test for session management vulnerabilities.

Last week we covered the following:

- Websites use sessions and cookies to deliver content that is specific to each visitor.
- After a user authenticates into a secure website, the web server issues the user a unique session cookie, so the information displayed is specific just to that user.

Session Hijacking

While the intended purpose of using session cookies is to keep a state between web pages when a user accesses a web application, a malicious user can obtain another user's unique session cookie and hijack the victim's private session.

This in an example of **session hijacking**.

Remember, we conducted a session hijacking attack when we used the Chrome browser extension Cookie-Editor to swap sessions.

Session Hijacking

Session hijacking can be conducted by several methods:

01

Sniffing Traffic

If a malicious user can sniff encrypted traffic, then they can potentially capture the session cookie and take over a victim's session.

02

Client-Side Attacks

A malicious user can deploy a cross-site scripting attack to steal a user's session cookie.

03

Predictable Sessions

A malicious user can predict what a unique session cookie might be.

Predictable Session Scenario

Henry, a malicious user, is using a stock-trading website to buy and sell stocks and mutual funds.

Henry logs in with his own credentials on Monday, February 9th, 2021, to the stock-trading site, then checks his session cookie in his browser settings:

`020921MON-1454`

He logs out and immediately logs back in again with his own credentials, then checks his session cookie again:

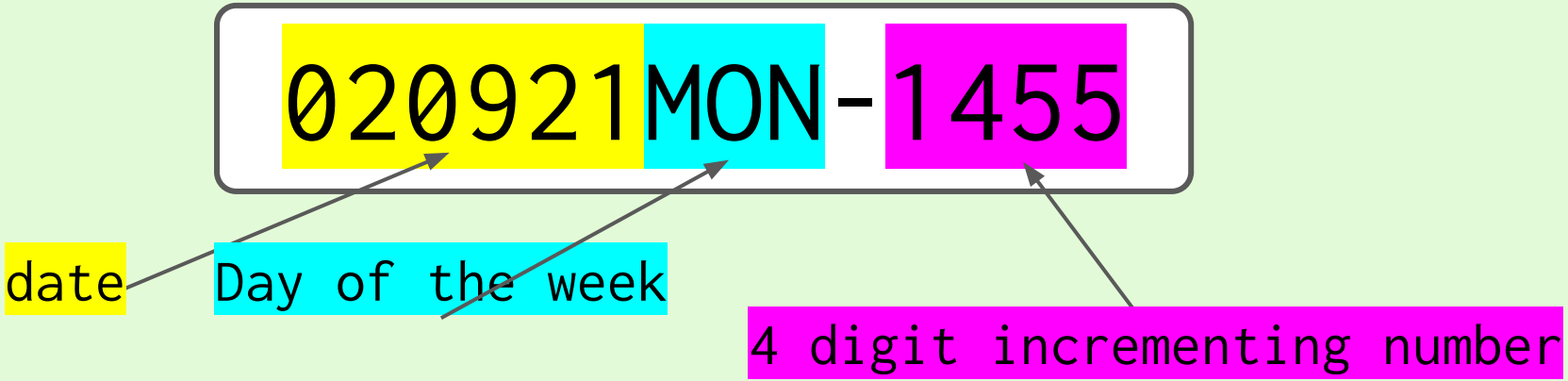
`020921MON-1455`

Henry logs in the following day, Tuesday, February 10th, 2021, and his session cookie is now:

`021021TUE-3834`

Predictable Session Scenario

Henry can review these session cookies that are generated and determine that the algorithm used by the stock trading site likely comprises the following:



Predictable Session Scenario

By figuring out the algorithm, Henry can try and guess another user's session cookie, then hijack another user's session.

Henry would first try `021121WED-1111`

If that didn't work, he would try `021121WED-1112`

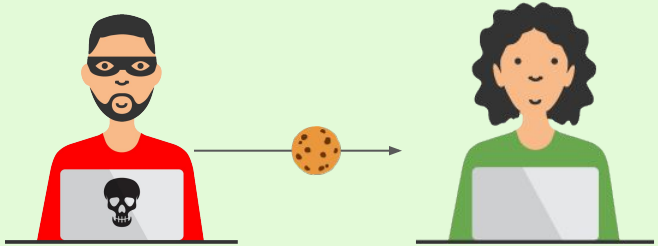
If that didn't work, he would try `021121WED-1113`

If that didn't work, he would try `021121WED-1114`

Eventually, after many tries, Henry tries a session cookie of

`021121WED-1118`

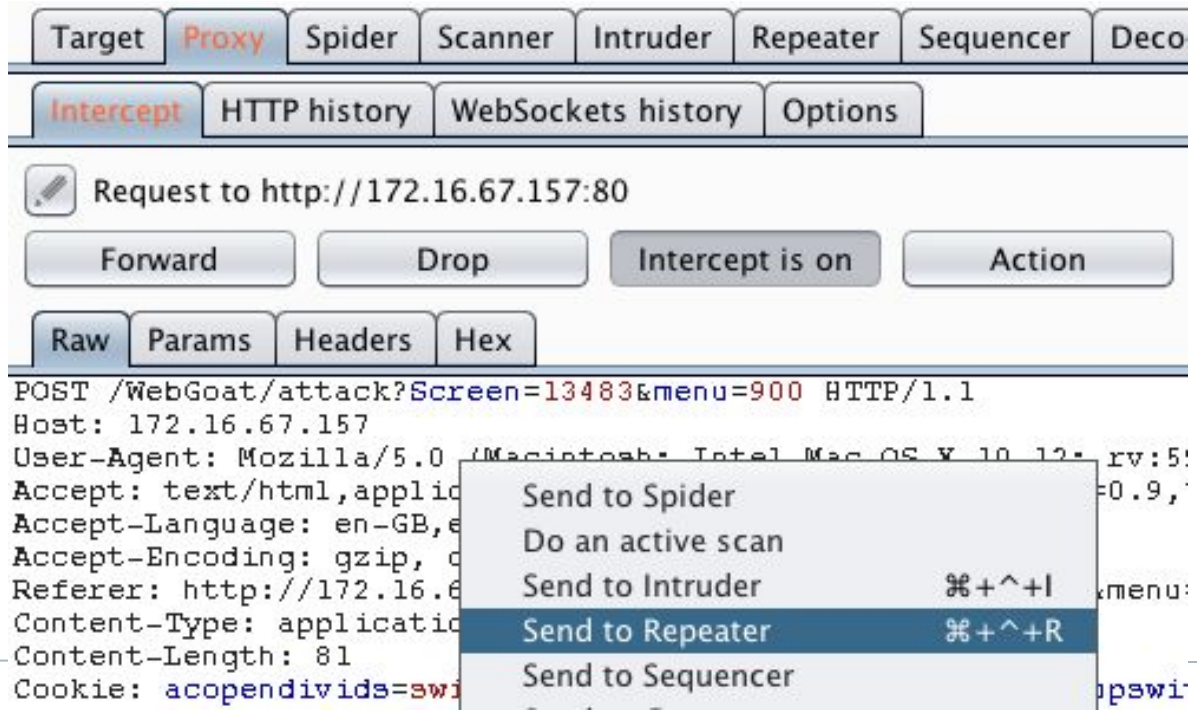
and then he ends up inside the stock-trading account of another user, Julie Jones.



Analyzing Session Management Vulnerabilities with Burp Repeater

Burp Suite Repeater

Burp Suite Repeater can simplify the process detailed in the previous scenario by using an algorithm to generate session cookies.



Burp Repeater Demo

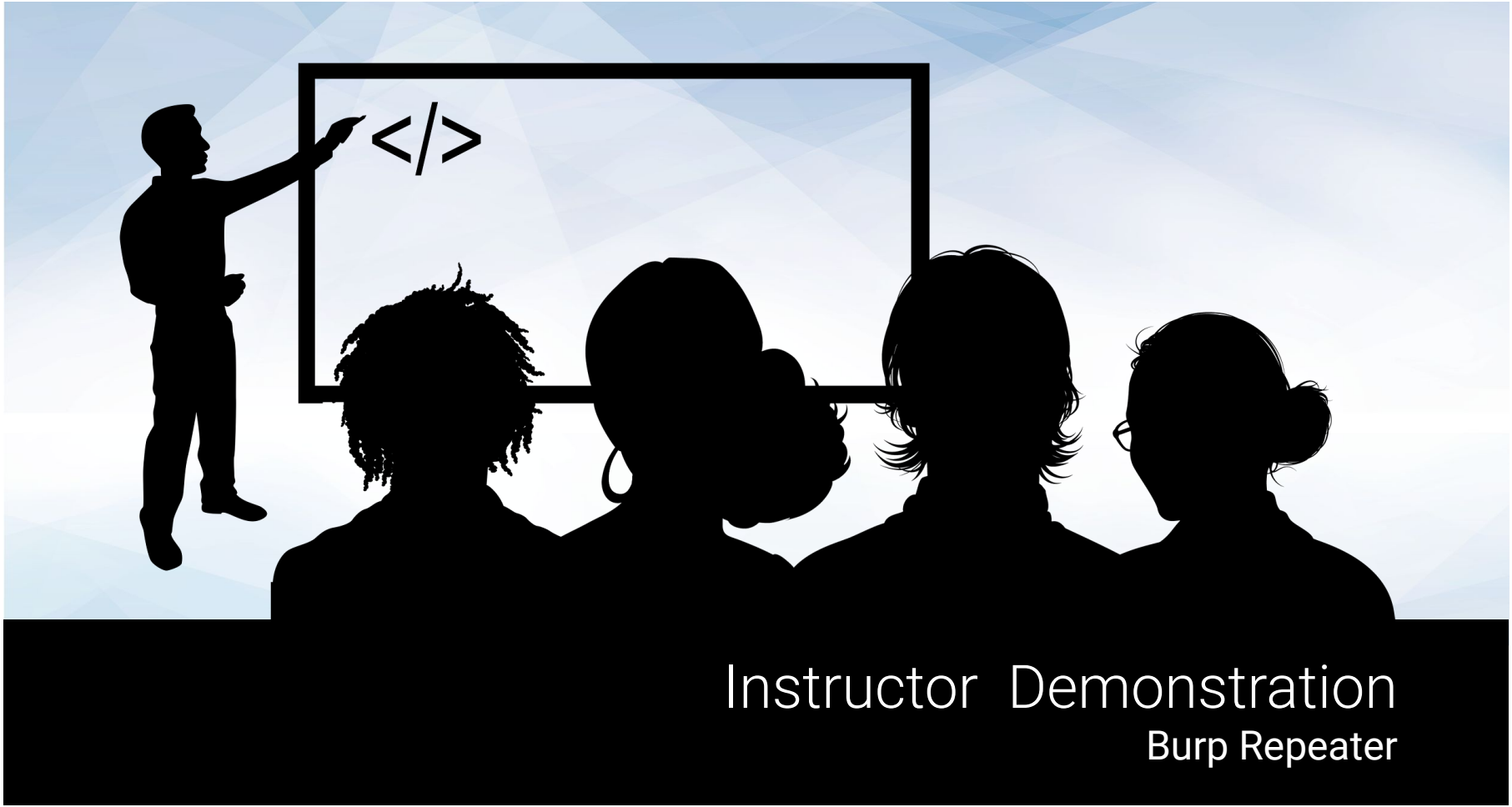
In this demonstration, we will continue to work on the Replicants web application.

01 Access the session cookie generator and enable proxy settings.

02 Generate and review the session cookie.

03 Move the HTTP request to Burp Repeater.

04 Use Burp Repeater to review the HTTP response.



Instructor Demonstration

Burp Repeater

Real-World Challenges, Impact, and Mitigations

Real-World Challenges

While the purpose of the lesson was to illustrate how to determine session management vulnerabilities with the Burp Repeater feature, in the real world we need to be aware of several issues not covered in this class.

Generation of session cookies

- While we clicked a button to generate session cookies, real applications do not have a generate button.
- Most web applications generate the session cookie on successful authentication into the web application.
- Security professionals have to determine the HTTP request that generates the session cookie and use that request in Burp Repeater.

Use of Burp repeater

- While we used Burp Repeater to simply reissue the same HTTP request, Burp Repeater can be used to modify the request one at a time.
- On the HTTP Request panel, an attacker can simply change any data in the request to check how it changes the response.
- Attackers can used this to try attacks against different hosts, resources, and types of HTTP requests.

Impact

If a malicious actor can successfully launch a session hijacking against a web application's users, they can potentially do the following:

- Access a user's confidential data
- Use features that are not intended to be accessed by unauthorized users within a secure application.

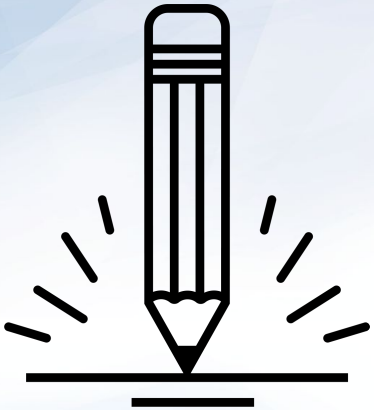
Mitigation Methods

Session hijacking mitigation varies depending on execution method:

Sniffing Attacks Mitigation	The web developer of the application can use an HTTP header to have a Secure cookie attribute. This ensures that the client only sends the session cookie encrypted and cannot be exposed in the web traffic.
Client Side Attacks	The web developer of the application can use an HTTP header to have an HttpOnly cookie attribute. This ensures that the client does not allow scripts, such as JavaScript, to access the session cookie. This can protect against a cross-site scripting attack trying to steal a cookie.
Predictable Sessions	web developer of the application can prevent the session cookie being generated from being predicted or brute forced, by ensuring the following: The cookie is long enough to make it difficult to brute force; the cookie is random enough to be unpredictable; and the cookie doesn't contain any details about the user.

Session Management Vulnerability Summary

Intended purpose of the original function	Session cookies are INTENDED to maintain a user's session within a web application.
Vulnerability, and method of exploit	With a session management vulnerability such as a predictable session, an attacker could predict a future session cookie and conduct a session hijacking attack.
Unintended consequence of the exploit	The unintended consequence is that if an attacker determines a victim's session cookie, they can access the victim's private session.
Mitigation of the vulnerability	Mitigations can vary depending on the session management vulnerability you are protecting against. Mitigations can include using protective HTTP headers and generating session cookies that are difficult to predict.
Potential impact of the exploit	The impact could include accessing a victim's confidential data inside the application or conducting unauthorized activities inside the application.



Activity: Analyzing Session Management Vulnerabilities with Burp Repeater

In this activity you will use Burp Suite to analyze session IDs that are generated from your application and determine whether they are randomly generated and thus at a risk of session hijacking.

Suggested Time:
20 Minutes

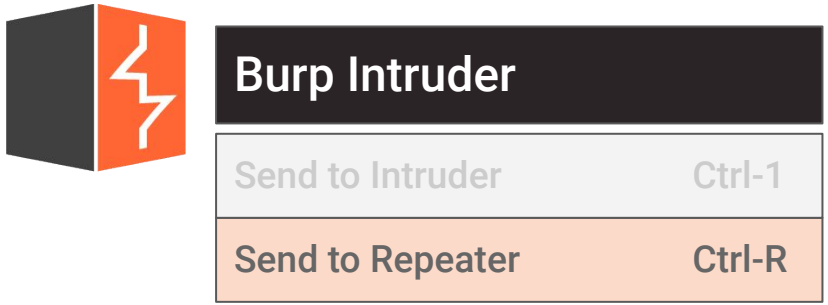
Time’s Up! Let’s Review.

Conducting Brute Force Attacks with Burp Intruder

Conducting Brute Force Attacks with Burp Intruder

Burp Repeater is limited to re-issuing *individual* HTTP messages.

Sometimes it is useful to issue multiple HTTP messages and automate multiple HTTP messages with a single request.



Scenario

An attacker is trying to attack a web application form field with a SQL injection attack.

The attacker attempts a SQL injection attack, with the common payload of:

```
' OR '1' = '1
```

Unfortunately for the attacker, the common payload

```
' OR '1' = '1
```

doesn't work.

The attacker now wants to try many different payloads, such as:

```
-- or #
```

```
' OR '1
```

```
' OR 1 -- -
```

```
" OR "" = "
```

```
" OR 1 = 1 -- -
```

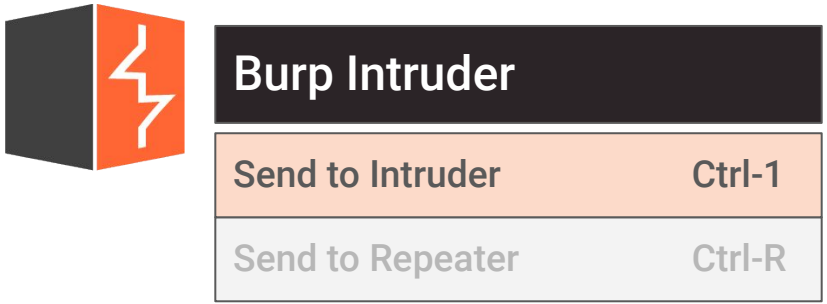
```
' OR '' = '
```

Rather than attempting and check each one at a time, it would be more efficient for the attacker to use a tool to try all of these payloads with a single request.

Burp Intruder

It also provides an understandable summary of the results of the different payloads.

Burp Intruder can automate issuing multiple HTTP messages.



Brute Force Attacks

Brute force attacks systematically attempt many passwords and username/password combinations with the hope of correctly determining the correct result to obtain access in an unauthorized area.

Credential Stuffing

Using lists of usernames to conduct brute force attacks that were obtained from breaches of other websites

Password Spraying

Using a single weak password, such as "123456", against a large list of usernames.

Brute force attacks also fall under the OWASP Top 10 risk of Identification and Authentication Failures (A07), as the attacker aims to bypass an application's method to securely authenticate a user.

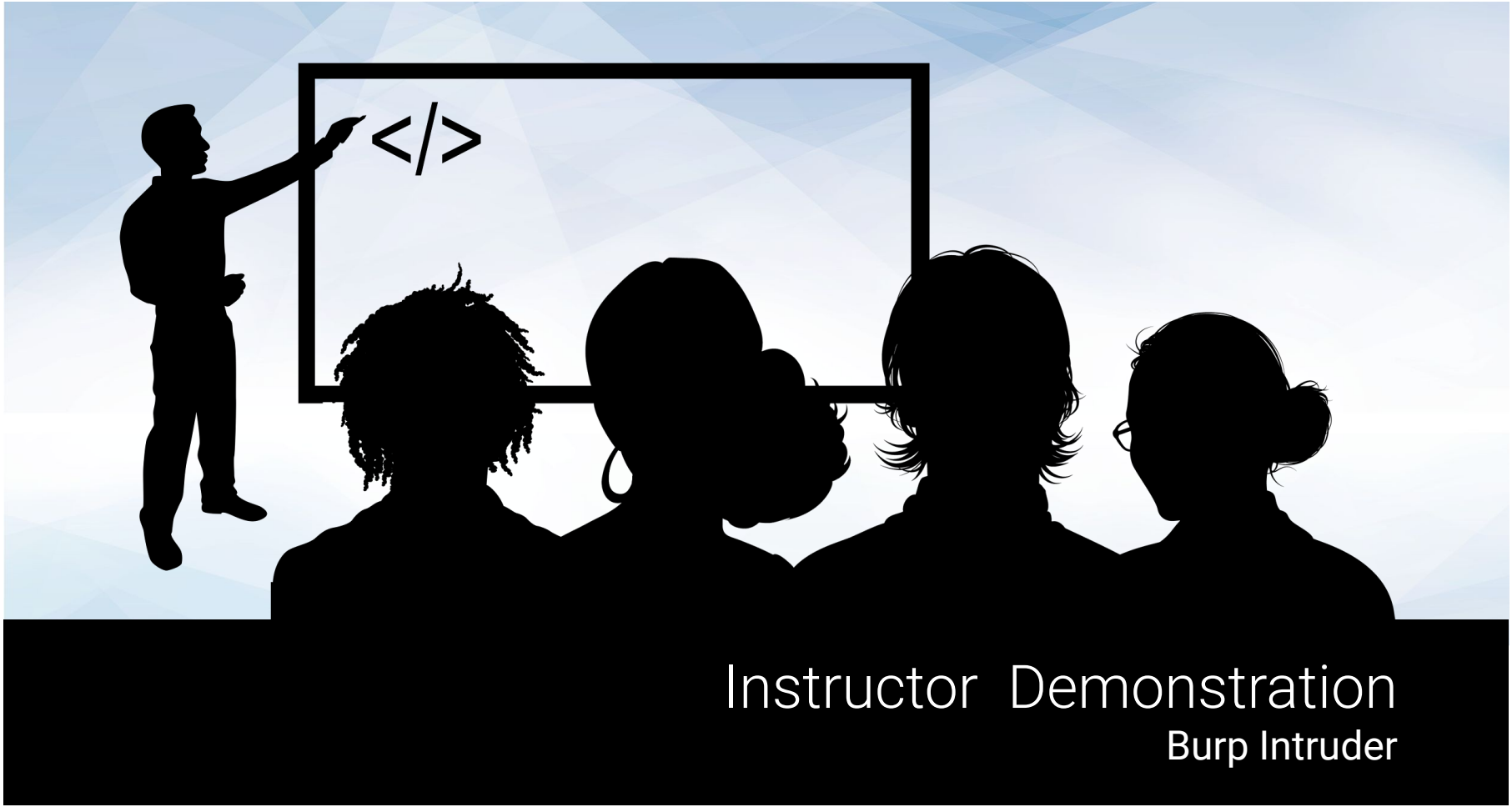
Brute Force Attacks

An attacker can use a brute force attack to continuously attempt username and password combinations.

Burp Intruder Demonstration

In the following demonstration, we will:

- Access the Replicants administrator login page and enable proxy settings.
- Generate and display a login request.
- Move the HTTP request to Burp Intruder.
- Identify the payload positions.
- Update the intruder payloads.
- Launch the brute force attack, and analyze the results.



Instructor Demonstration

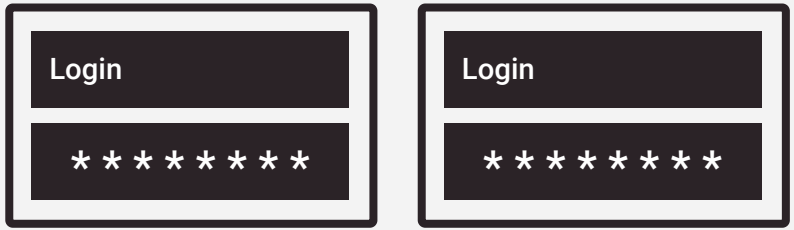
Burp Intruder

Real-World Challenges, Impact, and Mitigations

Real-World Challenges

Uses of Burp Intruder

While we used Burp Intruder to test a small list of two users and two passwords, most brute force tests would include much larger lists.



We used the Burp Intruder to conduct a brute force attack, but Burp Intruder can be used to conduct a variety of other web application attacks that test different payloads and payload combinations.

Impact

If a malicious actor can successfully launch a brute force attack against a web application's users and gain access to users' accounts, they can do the following:

- Access a user’s confidential data
- Use features that are not intended to be accessed by unauthorized users within a secure application

Mitigation Methods

Developers can use several mitigation methods to protect against a brute force attack:

01

Require complex usernames and passwords:

For example, require the user to include special characters, upper and lowercase, and numbers in the username and password.

02

Lock out accounts after a number of failed attempts:

For example, after three failed login attempts, the user's account gets locked.

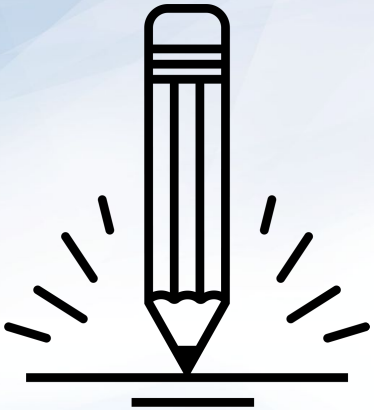
03

Use multi-factor authentication (MFA):

For example, require users to have a password and a secondary form of authentication, like a pin generated by an external token.

Brute Force Attack Summary

Intended purpose of the original function	Web applications use usernames and password input fields to authenticate a user before they can access a secure part of a web application.
Vulnerability, and method of exploit	Without proper protections on the login page, an attacker can apply a brute force attack to systematically test many user/password combinations to attempt to gain access to unauthorized accounts.
Unintended consequence of the exploit	The unintended consequence is that if an attacker can determine a correct username/password combination, they can access an account that they do not have permission to access.
Mitigation of the vulnerability	Mitigations can include requiring complex usernames and passwords, using multi-factored authentication, and enabling a lockout after a certain amount of failed login attempts.
Potential impact of the exploit	The impact could include accessing a victim's confidential data inside the application or conducting unauthorized activities inside the application.

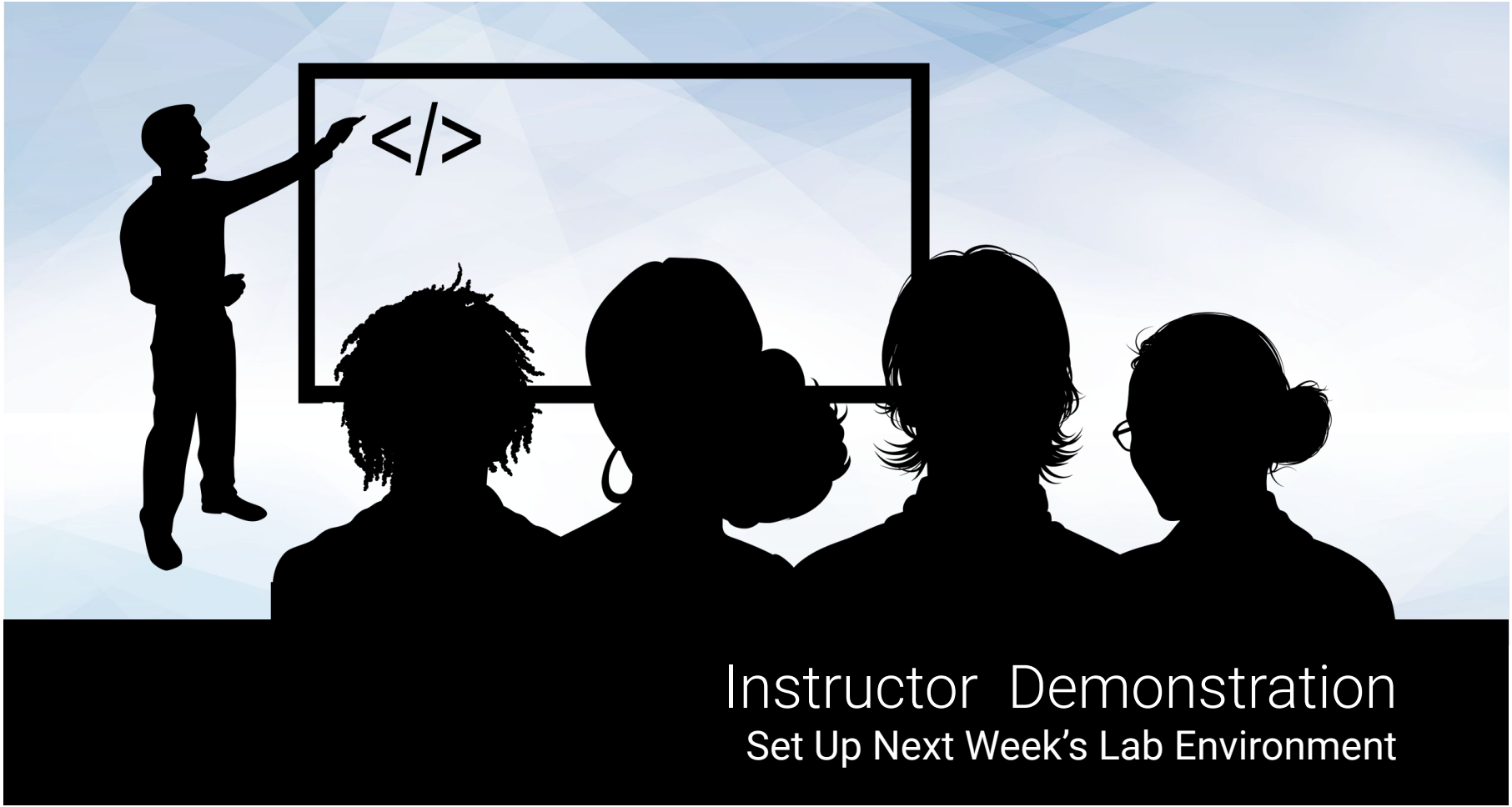


Activity: Brute Force Attacks with Burp Intruder

In this activity, you will use Burp Suite to determine if weak user passwords are vulnerable to brute force attacks.

Suggested Time:
20 Minutes

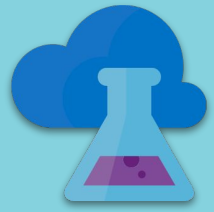
Time’s Up! Let’s Review.



Instructor Demonstration

Set Up Next Week's Lab Environment

Next Week's Lab Environment



Next week, we will return to Azure Lab Services and use a new Pentesting lab environment.

- ✓ Switch to your local computer environment.
- ✓ You will be provided with a registration link for the Pentesting environment.
- ✓ Once you click the link, the Pentesting environment card will be added to your Azure dashboard.
- ✓ Make sure you can set up and access this environment. Reach out if you need help with any troubleshooting issues prior to the next class.


The End.