CEH v12 Lesson 7 : Web Application Exploitation Attacks

Part 1

Learning Outcomes

In this module, you will complete the following exercises:

- Exercise 1 — Attacking Authentication Mechanisms
- Exercise 2 — Attack Authorization Schemes
- Exercise 3 — Attack Access Controls
- Exercise 4 — Attack Session Management Mechanisms

After completing this module, you will be able to:

- Attack Authentication Mechanisms Using Brutespray
- Perform Authorization Attack Using URI
- Conduct Parameter-based Access Control Attack
- View Session ID and launch another session

Lab Duration

It will take approximately **1 hour** to complete this lab.

Exercise 1 — Attacking Authentication Mechanisms

An attacker can perform different types of authentication attacks on a target, which are possible because of the design flaws within a web application. Commonly exploited flaws include:

- Predictable usernames
- Cookie poisoning
- Cookie sniffing

- Password guessing
- Brute-force attack
- Dictionary attack
- Session poisoning

An attacker needs to find just one of these design flaws and exploit them.

In this exercise, you will attack an authentication mechanism.

Learning Outcomes

After completing this exercise, you will be able to:

- Attack Authentication Mechanism Using Brutespray

Exercise 2 — Attack Authorization Schemes

In the initial stages of an attack on a web application, an attacker is using what is known as a low-privileged attack, but during the main phases of an attack, they will escalate privileges to access the resources that would have not been accessible otherwise.

An attacker can use various methods to conduct an authorization attack, including the following methods:

- Query String Tampering
- HTTP Header Tampering
- Parameter Tampering
- Uniform Resource Identifier (URI) Usage
- Hidden Tags Manipulation

In this exercise, you will learn about the authorization attack through URI usage.

Learning Outcomes

After completing this exercise, you will be able to:

- Perform Authorization Attack Using URI

Your Devices

You will be using the following devices in this lab. Please power these on now.

PLABDC01Domain Controller192.168.0.1/24PLABWIN10Domain MemberWorkstation192.168.0.3/24PLABKALI01Domain MemberWorkstation192.168.0.5/24

- PLABDC01

Windows Server 2019 — Domain Server192.168.0.1/24

- PLABWIN10

Windows 10 — Workstation192.168.0.3/24

- PLABKALI01

Kali 2022.1 — Linux Kali Workstation192.168.0.5/24

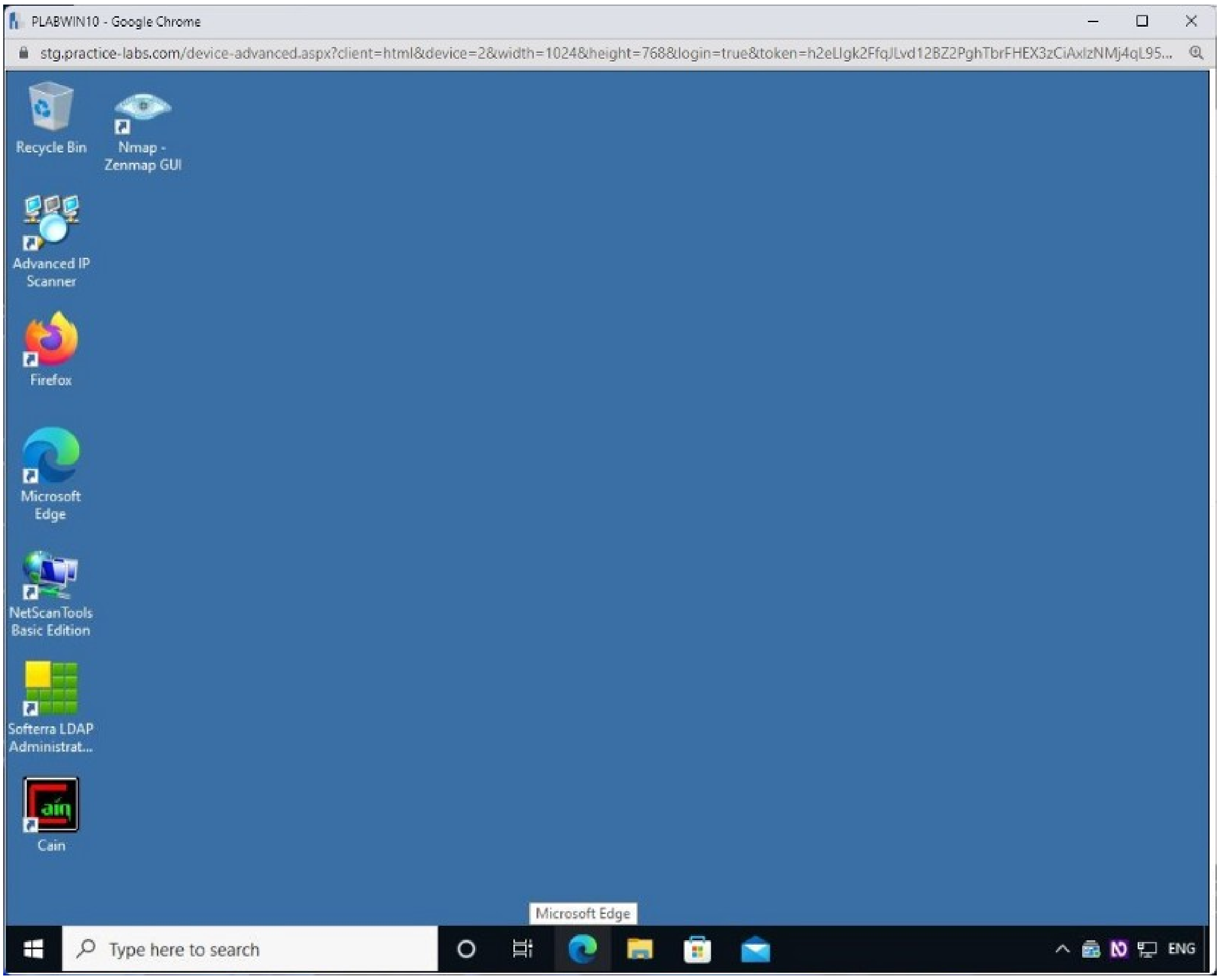
Task 1 — Perform Authorization Attack Using URI

In a web application, you need to ensure that you restrict access to certain resources that should not be accessible to everyone. For example, certain documents should not be accessible publicly but only to the individuals who have logged on to a web application. If an attacker can bypass the login and access the documents, then it means that an authorization attack has occurred.

In this task, you will learn to perform an authorization attack using URI.

Step 1

Connect to **PLABWIN10**. Click the **Microsoft Edge** icon in the taskbar.

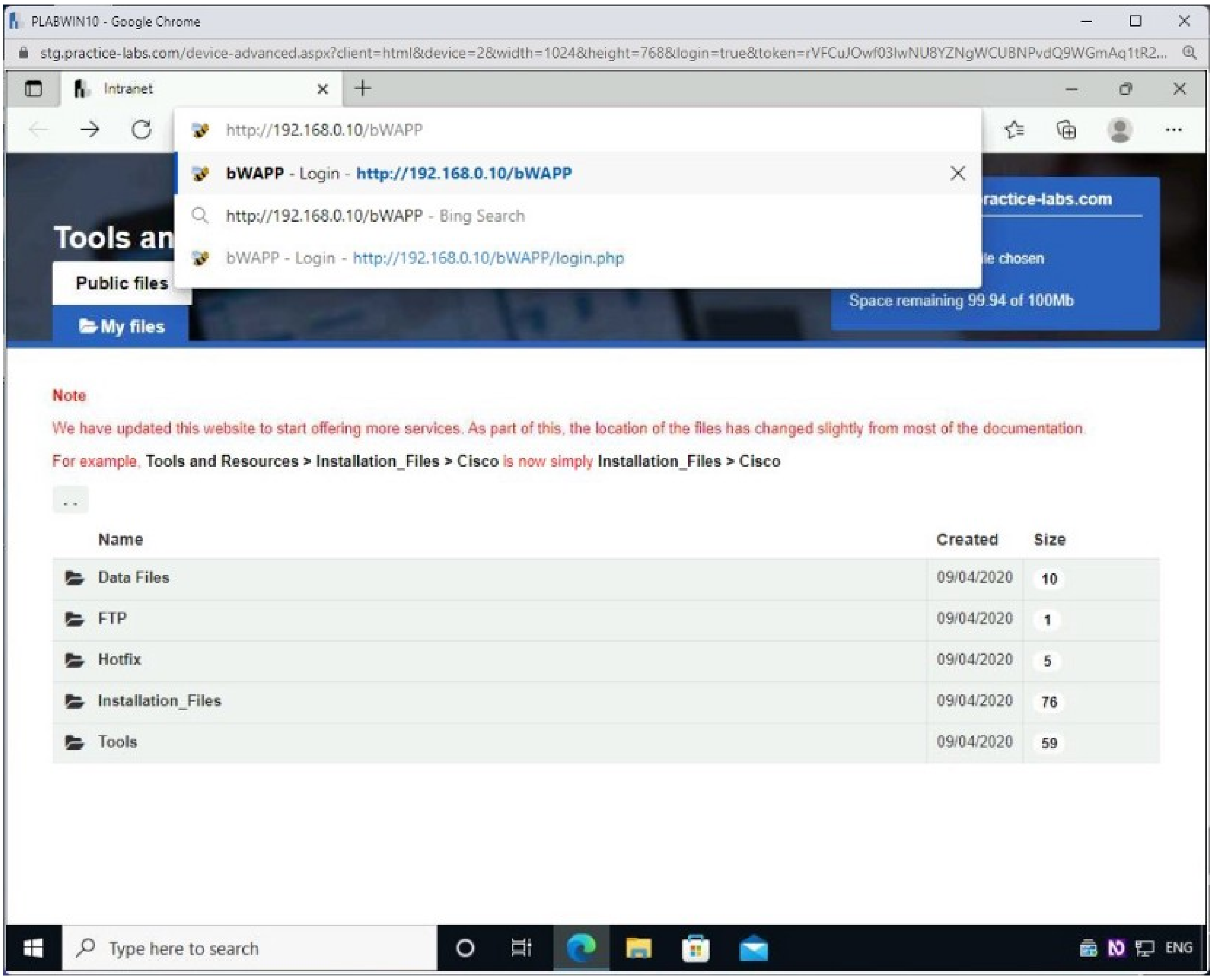

PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=h2eLlgk2FfqJLvd12BZ2PghTbrFHEX3zCiAxlzNMj4qL95...
Recycle Bin
Nmap - Zenmap GUI
Advanced IP Scanner
Firefox
Microsoft Edge
NetScanTools Basic Edition
Softerra LDAP Administrat...
Cain
Microsoft Edge
Type here to search
ENG

Step 2

To access the **bWAPP** application in the **Microsoft Edge** window, type the following URL in the address bar:

```
http://192.168.0.10/bWAPP
```

Press **Enter**.


PLABWIN10 - Google Chrome

stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03IwNU8YZNgWCUBNPvdQ9WGmAq1tR2...

Intranet

http://192.168.0.10/bWAPP

bWAPP - Login - http://192.168.0.10/bWAPP

http://192.168.0.10/bWAPP - Bing Search

bWAPP - Login - http://192.168.0.10/bWAPP/login.php

Note

We have updated this website to start offering more services. As part of this, the location of the files has changed slightly from most of the documentation.

For example, Tools and Resources > Installation_Files > Cisco is now simply Installation_Files > Cisco

Name	Created	Size
Data Files	09/04/2020	10
FTP	09/04/2020	1
Hotfix	09/04/2020	5
Installation_Files	09/04/2020	76
Tools	09/04/2020	59

Step 3

The login page of the bWAPP application is displayed.

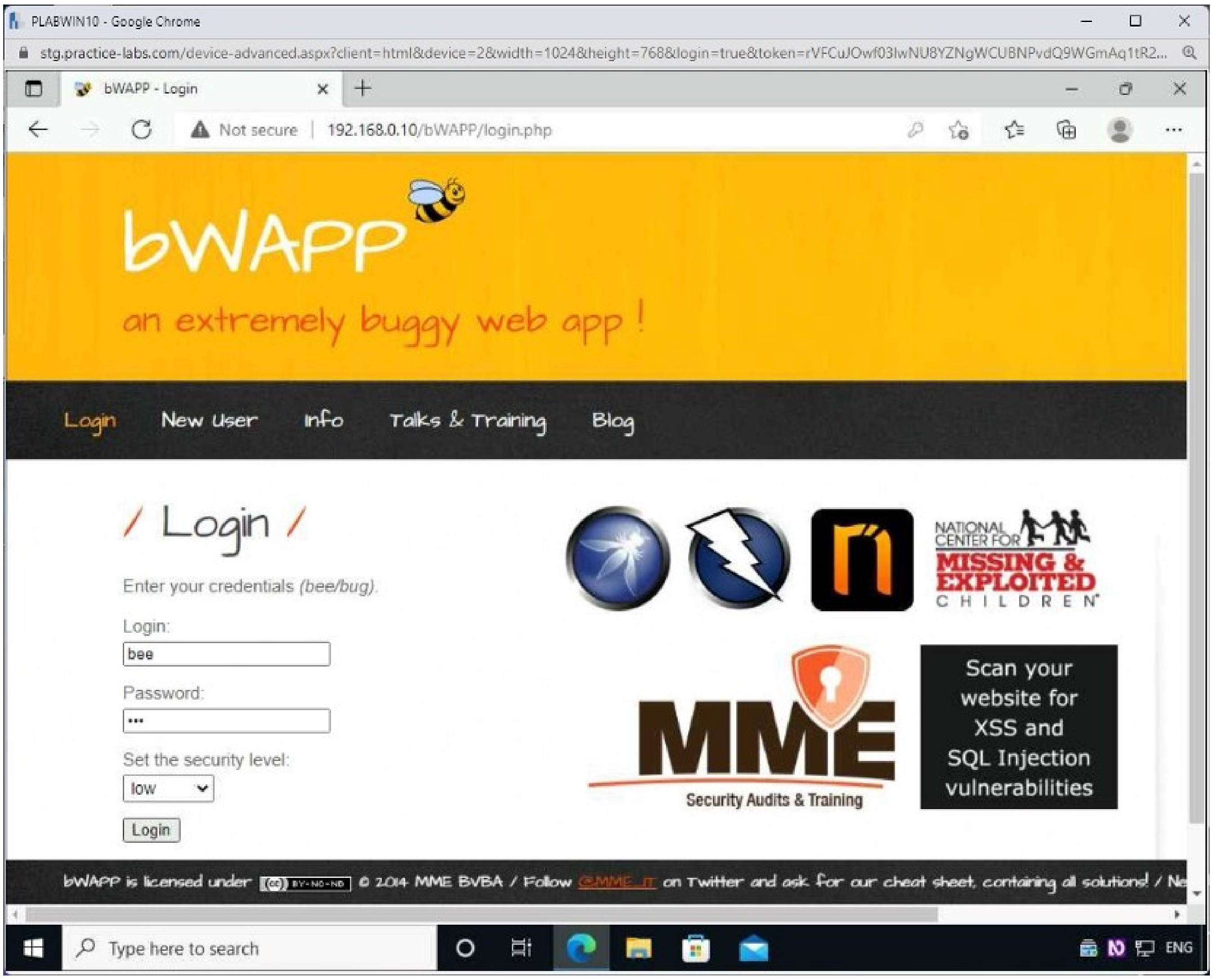
In the **Username** box on the bWAPP login page, type the following username:

```
bee
```

In the **Password** box, type the following password:

bug

Click **Login**.


PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03IwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Login
Not secure | 192.168.0.10/bWAPP/login.php
bWAPP
an extremely buggy web app !
Login
New User
Info
Talks & Training
Blog
/ Login /
Scan your website for XSS and SQL Injection vulnerabilities
MME Security Audits & Training
Type here to search
ENG

Enter your credentials (bee/bug).

Login: bee

Password: •••

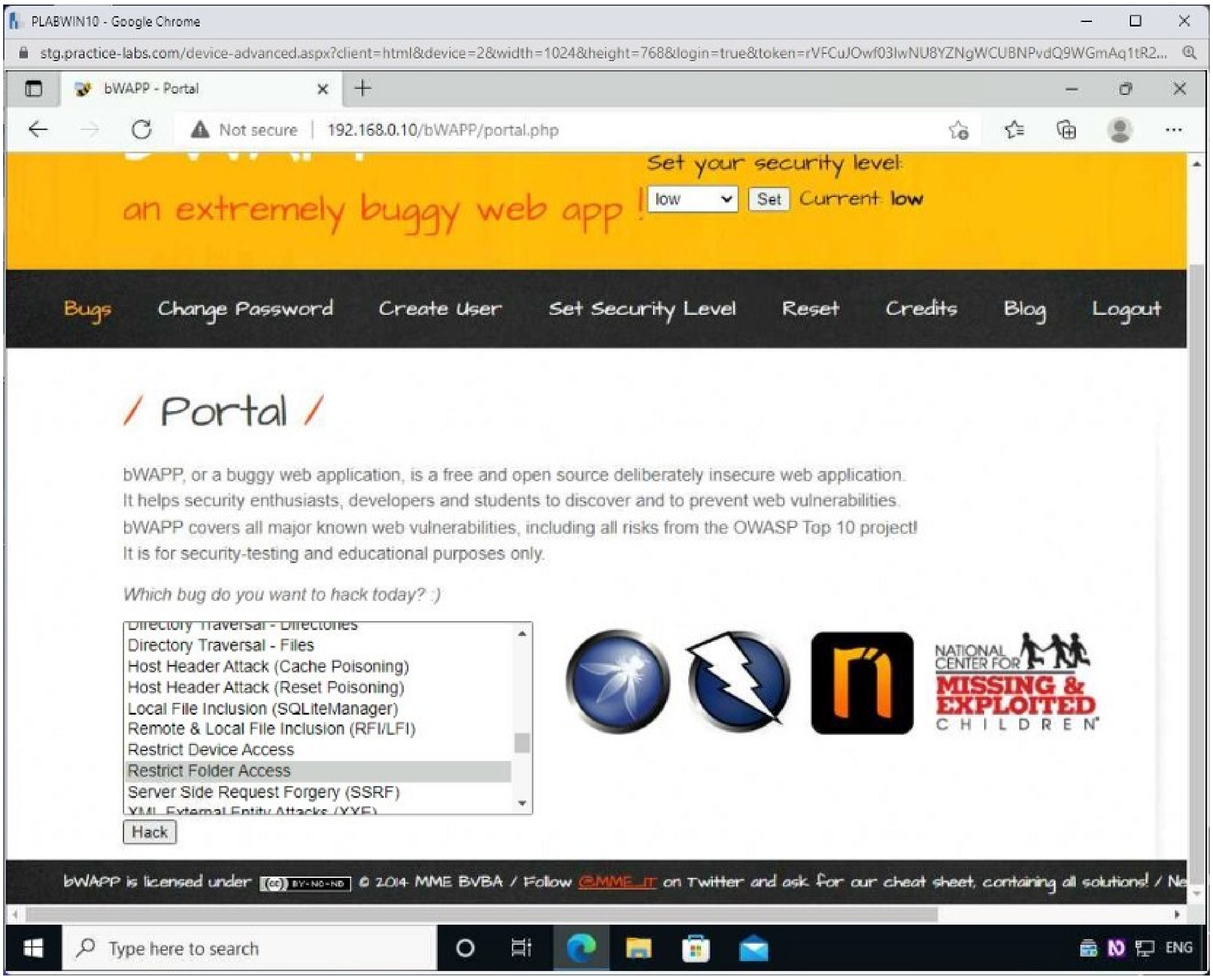
Set the security level: low

Login

bWAPP is licensed under (cc) BY-NC-ND © 2014 MME BVBA / Follow @MME_IT on Twitter and ask for our cheat sheet, containing all solutions! / Ne

Step 4

From the **Choose your bug** drop-down, select **Restrict Folder Access,** and click **Hack**.


PLABWIN10 - Google Chrome

stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03IwNU8YZNgWCUBNPvdQ9WGmAq1tR2...

bWAPP - Portal

Not secure | 192.168.0.10/bWAPP/portal.php

an extremely buggy web app !

Set your security level: low [Set] Current: low

Bugs | Change Password | Create User | Set Security Level | Reset | Credits | Blog | Logout

/ Portal /

bWAPP, or a buggy web application, is a free and open source deliberately insecure web application.
It helps security enthusiasts, developers and students to discover and to prevent web vulnerabilities.
bWAPP covers all major known web vulnerabilities, including all risks from the OWASP Top 10 project!
It is for security-testing and educational purposes only.

Which bug do you want to hack today? :)

- Directory Traversal - Directories
- Directory Traversal - Files
- Host Header Attack (Cache Poisoning)
- Host Header Attack (Reset Poisoning)
- Local File Inclusion (SQLiteManager)
- Remote & Local File Inclusion (RFI/LFI)
- Restrict Device Access
- Restrict Folder Access
- Server Side Request Forgery (SSRF)
- XML External Entity Attacks (XXE)

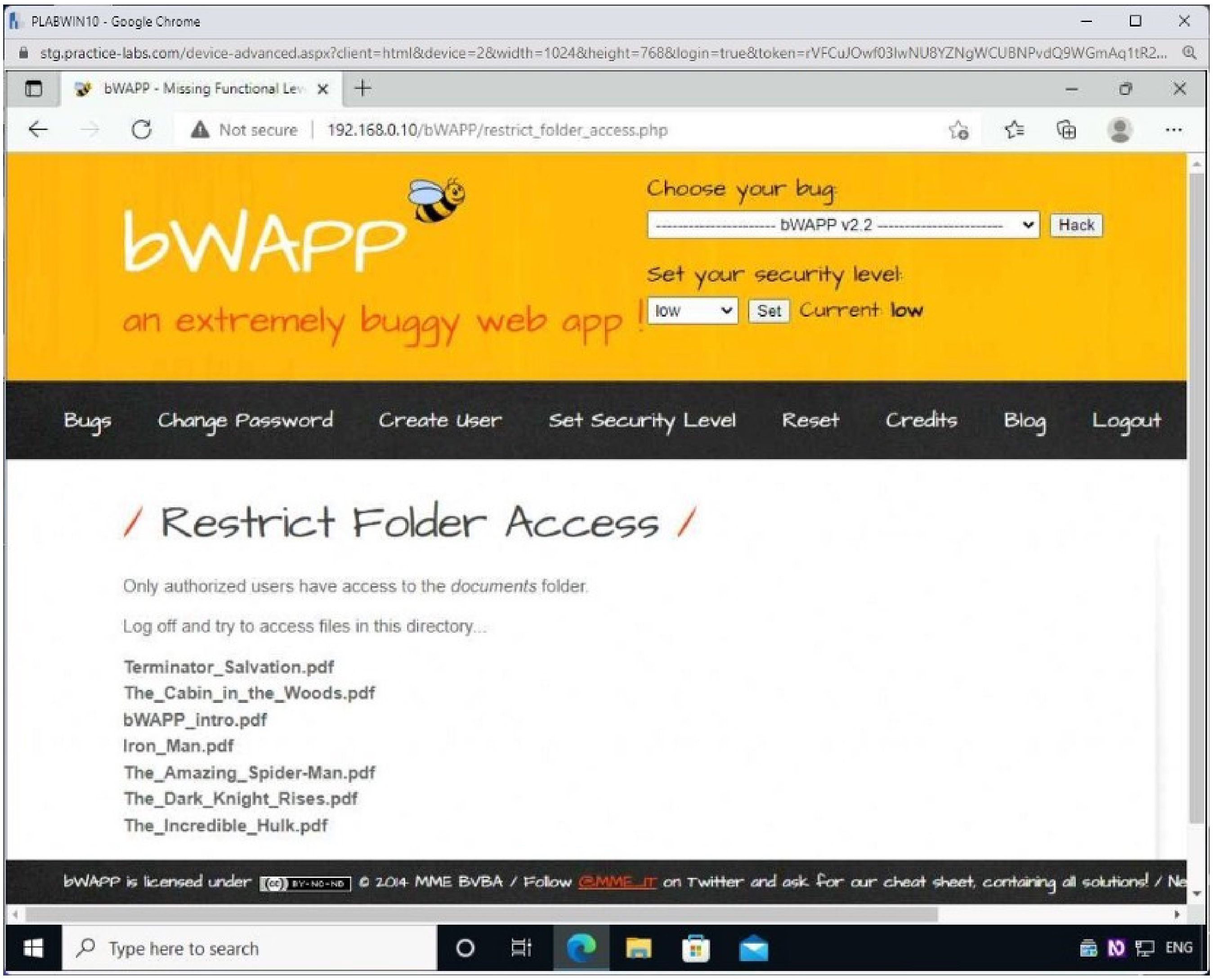
[Hack]

bWAPP is licensed under (cc) BY-NC-ND © 2014 MME BVBA / Follow @MME_IT on Twitter and ask for our cheat sheet, containing all solutions! / Ne

Step 5

Notice that there are several files located in this directory.

Click on the **bWAPP_intro.pdf** file.


PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03IwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Missing Functional Lev
Not secure
192.168.0.10/bWAPP/restrict_folder_access.php
bWAPP
an extremely buggy web app !
Choose your bug
bWAPP v2.2
Hack
Set your security level:
low
Set
Current: low
Bugs
Change Password
Create User
Set Security Level
Reset
Credits
Blog
Logout
Type here to search
ENG

/ Restrict Folder Access /

Only authorized users have access to the *documents* folder.

Log off and try to access files in this directory...

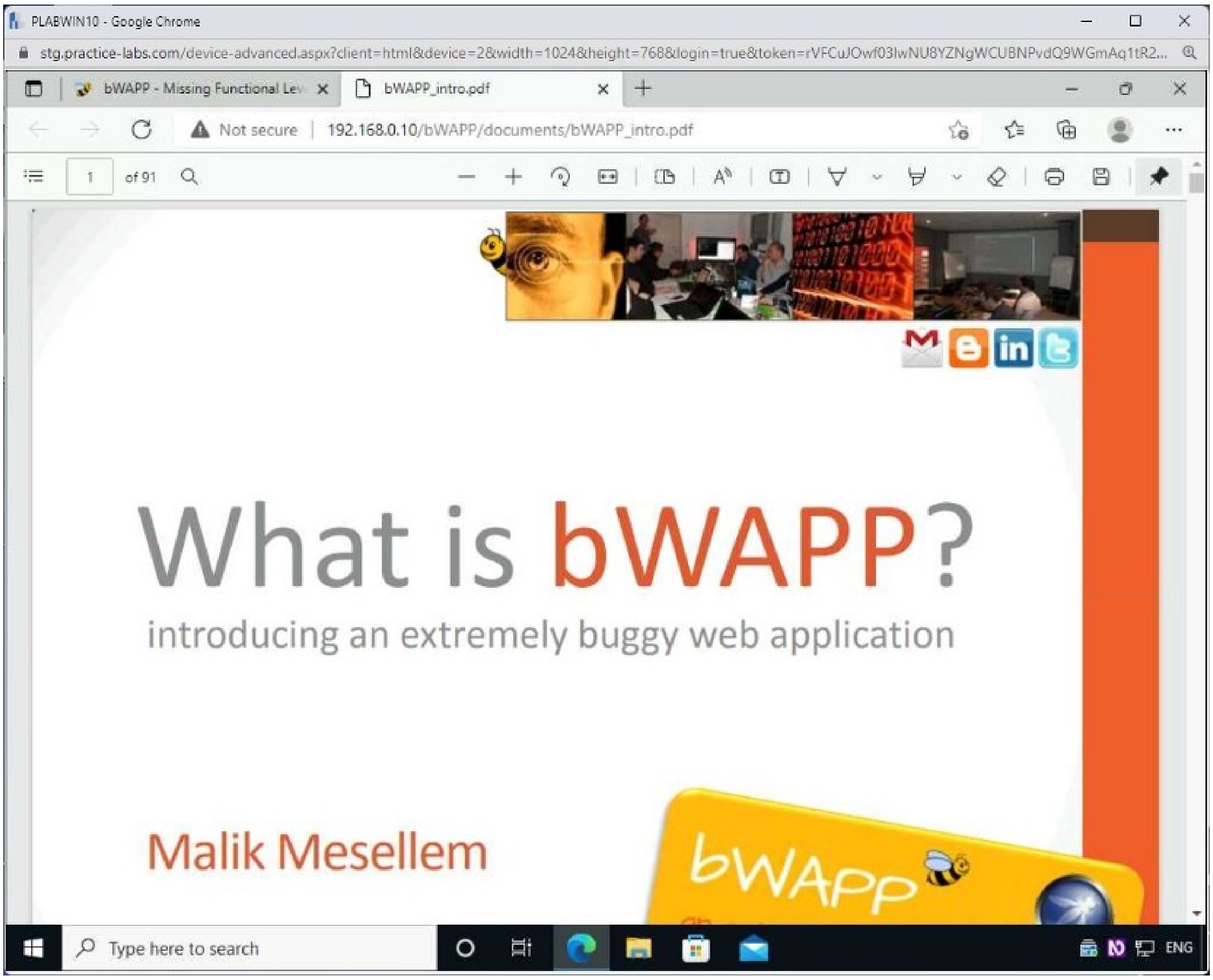
Terminator_Salvation.pdf
The_Cabin_in_the_Woods.pdf
bWAPP_intro.pdf
Iron_Man.pdf
The_Amazing_Spider-Man.pdf
The_Dark_Knight_Rises.pdf
The_Incredible_Hulk.pdf

bWAPP is licensed under (cc) BY-NC-ND © 2014 MME BVBA / Follow @MME_IT on Twitter and ask for our cheat sheet, containing all solutions! / Ne

Step 6

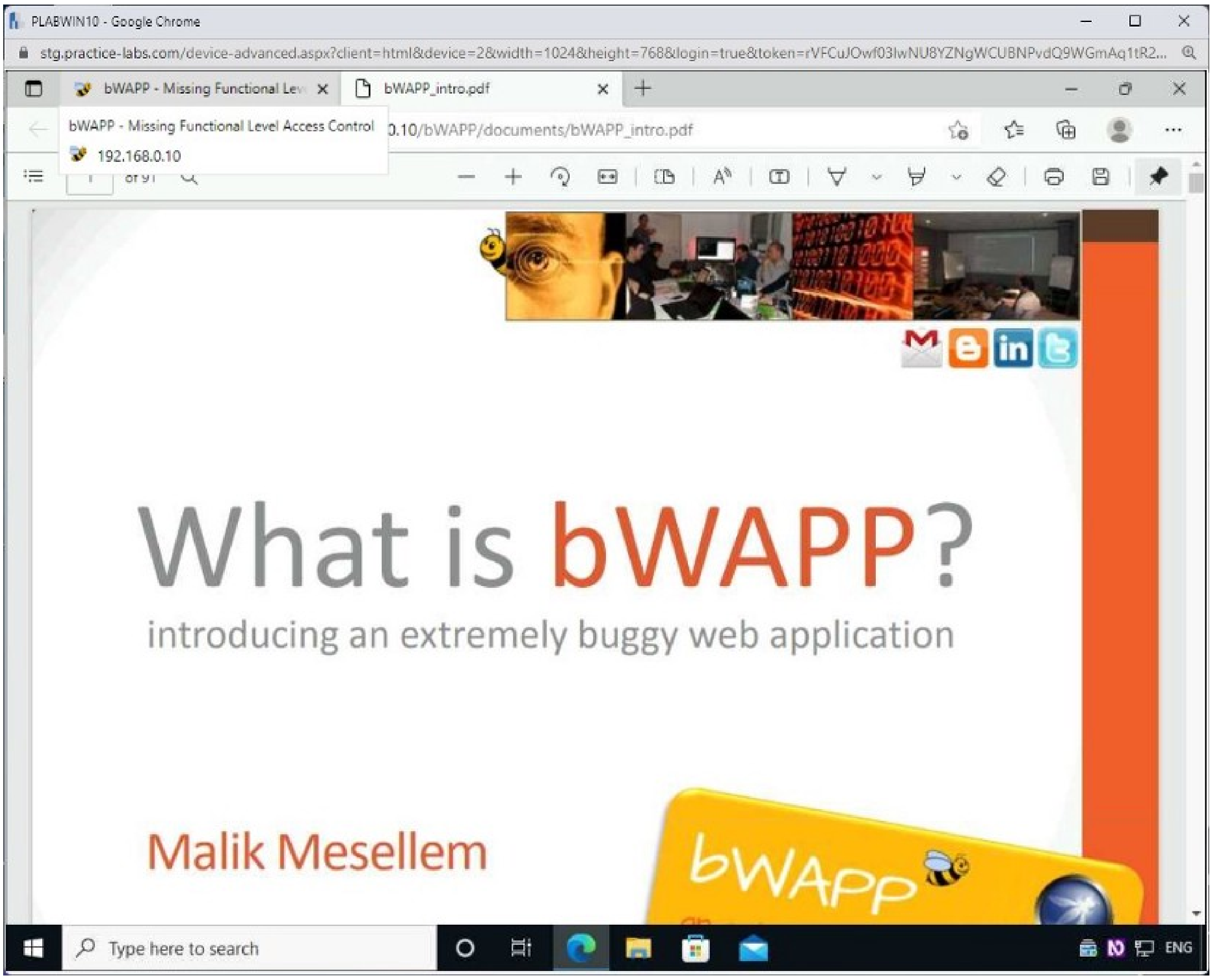
As the file opens, you will get the path including the directory name in the address bar in a new tab.

Notice that the directory named **documents**.


PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03lwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Missing Functional Lev
bWAPP_intro.pdf
Not secure
192.168.0.10/bWAPP/documents/bWAPP_intro.pdf
1
of 91
What is bWAPP?
introducing an extremely buggy web application
Malik Mesellem
bWAPP
Type here to search
ENG

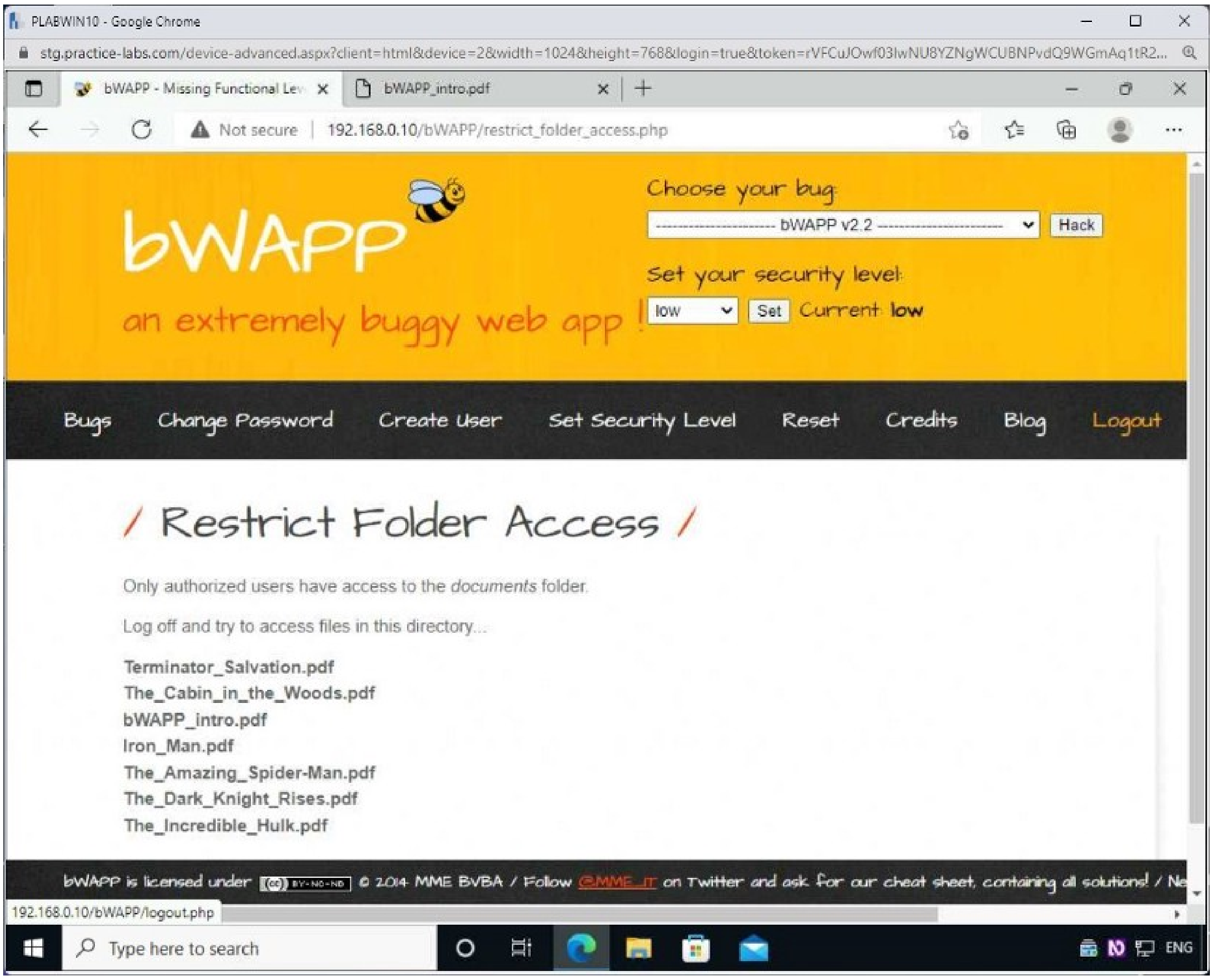
Step 7

Click the **bWAPP** tab.


PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03IwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Missing Functional Lev
bWAPP_intro.pdf
bWAPP - Missing Functional Level Access Control
192.168.0.10
0.10/bWAPP/documents/bWAPP_intro.pdf
of 91
What is bWAPP?
introducing an extremely buggy web application
Malik Mesellem
bWAPP
Type here to search
ENG

Step 8

On the **bWAPP** tab, click **Logout**.


PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03lwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Missing Functional Lev
bWAPP_intro.pdf
Not secure
192.168.0.10/bWAPP/restrict_folder_access.php
bWAPP
an extremely buggy web app !
Choose your bug:
bWAPP v2.2
Hack
Set your security level:
low
Set
Current: low
Bugs
Change Password
Create User
Set Security Level
Reset
Credits
Blog
Logout
bWAPP is licensed under © 2014 MME BVBA / Follow @MME_IT on Twitter and ask for our cheat sheet, containing all solutions! / Ne
192.168.0.10/bWAPP/logout.php
Type here to search
ENG

/ Restrict Folder Access /

Only authorized users have access to the *documents* folder.

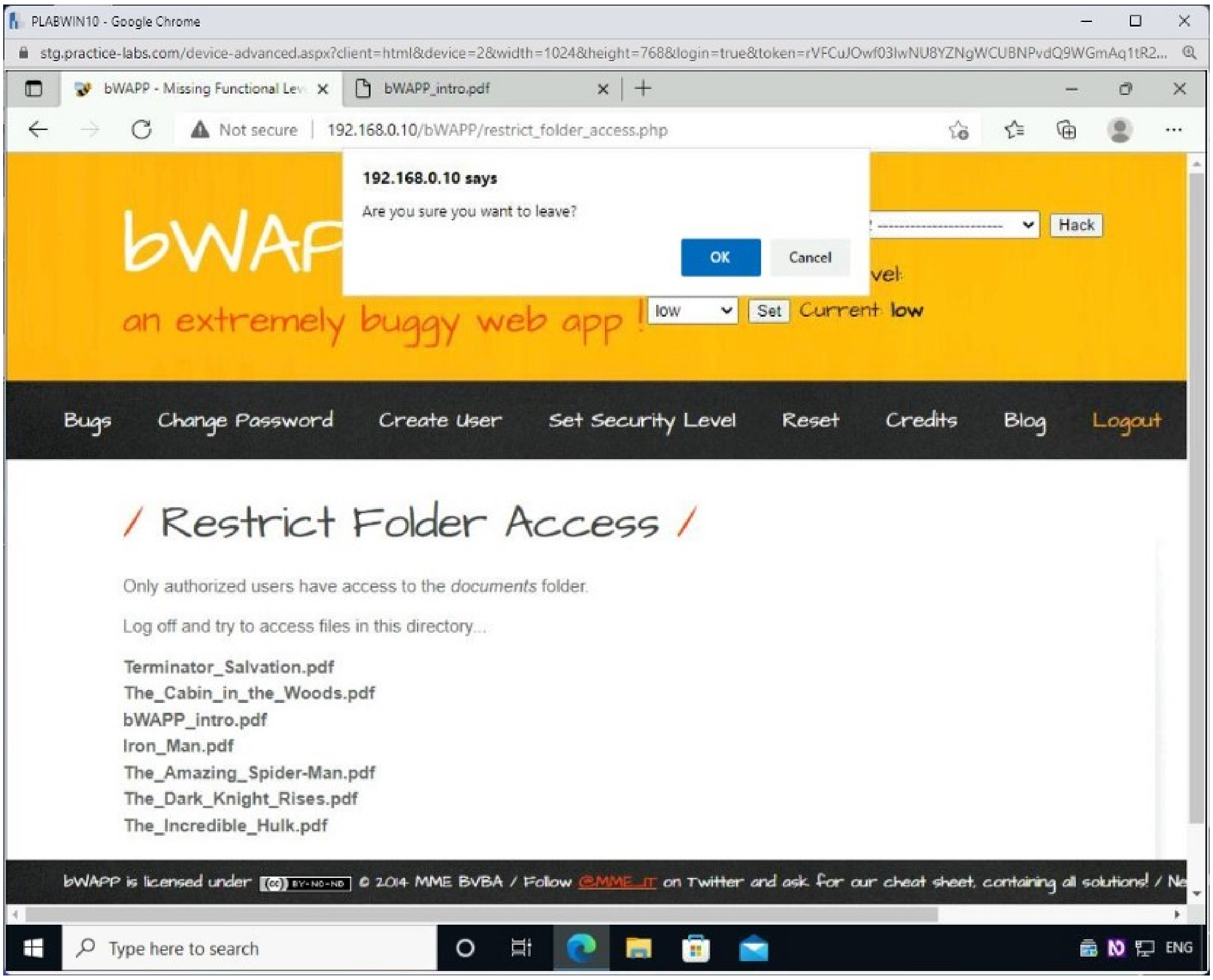
Log off and try to access files in this directory...

Terminator_Salvation.pdf
The_Cabin_in_the_Woods.pdf
bWAPP_intro.pdf
Iron_Man.pdf
The_Amazing_Spider-Man.pdf
The_Dark_Knight_Rises.pdf
The_Incredible_Hulk.pdf

Step 9

A dialog box is displayed confirming the logout.

Click **OK**.


192.168.0.10 says

Are you sure you want to leave?

/ Restrict Folder Access /

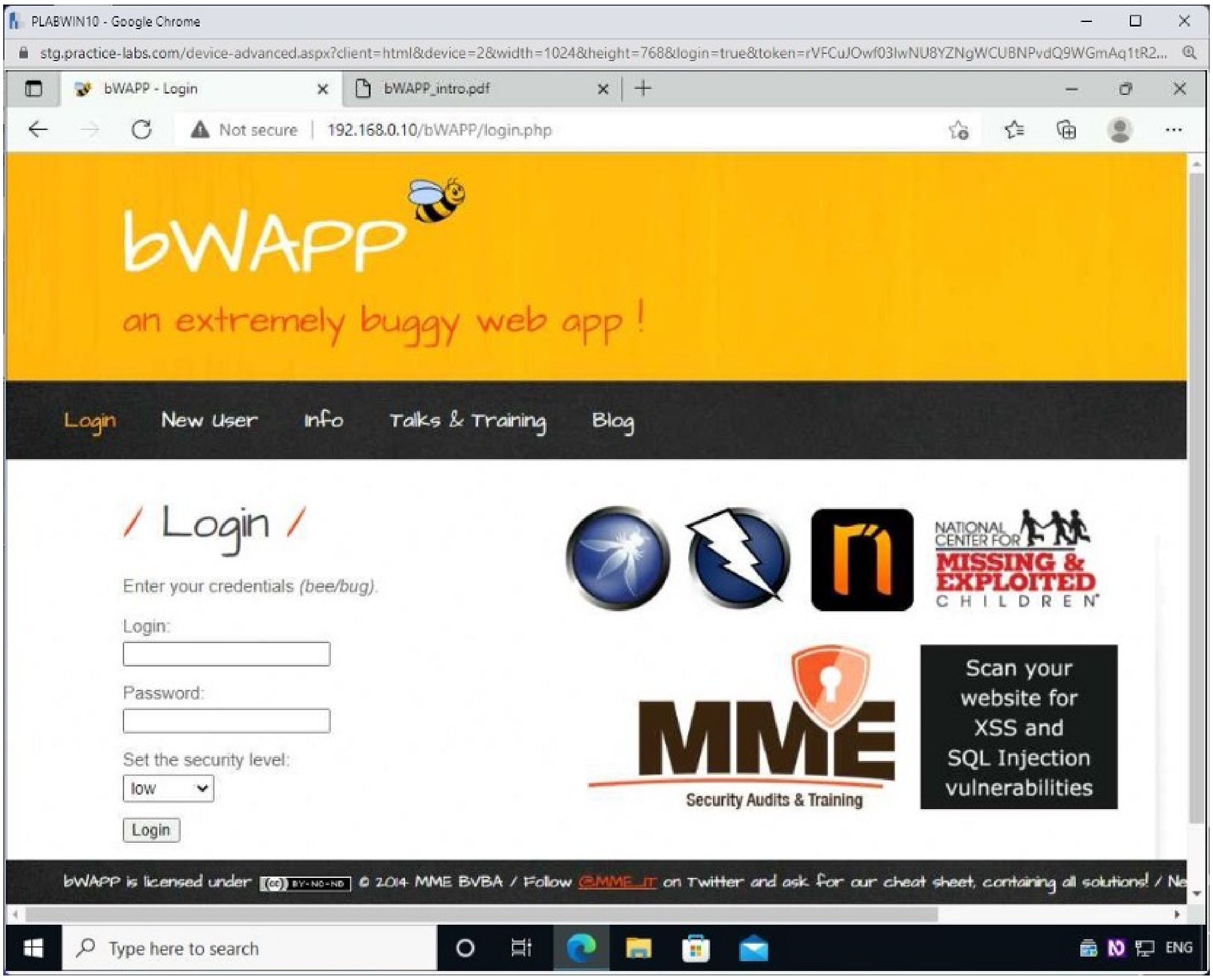
Only authorized users have access to the *documents* folder.

Log off and try to access files in this directory...

Terminator_Salvation.pdf
The_Cabin_in_the_Woods.pdf
bWAPP_intro.pdf
Iron_Man.pdf
The_Amazing_Spider-Man.pdf
The_Dark_Knight_Rises.pdf
The_Incredible_Hulk.pdf

Step 10

You are on the login page.


PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03IwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Login
bWAPP_intro.pdf
Not secure
192.168.0.10/bWAPP/login.php
bWAPP
an extremely buggy web app !
Login
New User
Info
Talks & Training
Blog
MME
Security Audits & Training
Scan your website for XSS and SQL Injection vulnerabilities
Type here to search
ENG

/ Login /

Enter your credentials *(bee/bug)*.

Login:

Password:

Set the security level:

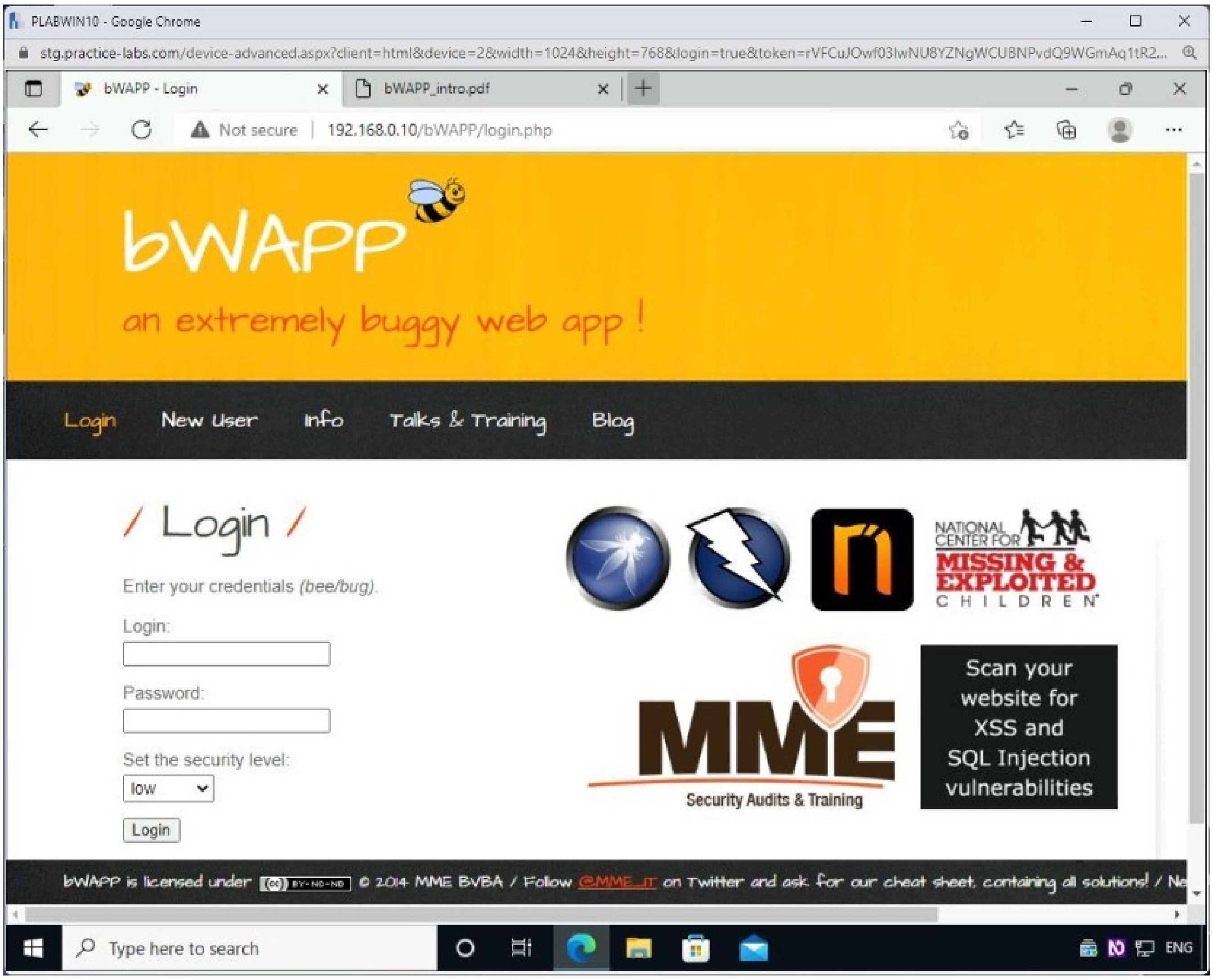
low

Login

bWAPP is licensed under © 2014 MME BVBA / Follow @MME_IT on Twitter and ask for our cheat sheet, containing all solutions! / Ne

Step 11

Click on the **+** (**New tab**) sign to open a new tab.


PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03IwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Login
bWAPP_intro.pdf
Not secure
192.168.0.10/bWAPP/login.php
bWAPP
an extremely buggy web app !
Login
New User
Info
Talks & Training
Blog
Scan your website for XSS and SQL Injection vulnerabilities
Type here to search
ENG

/ Login /

Enter your credentials *(bee/bug)*.

Login:

Password:

Set the security level:

low

Login

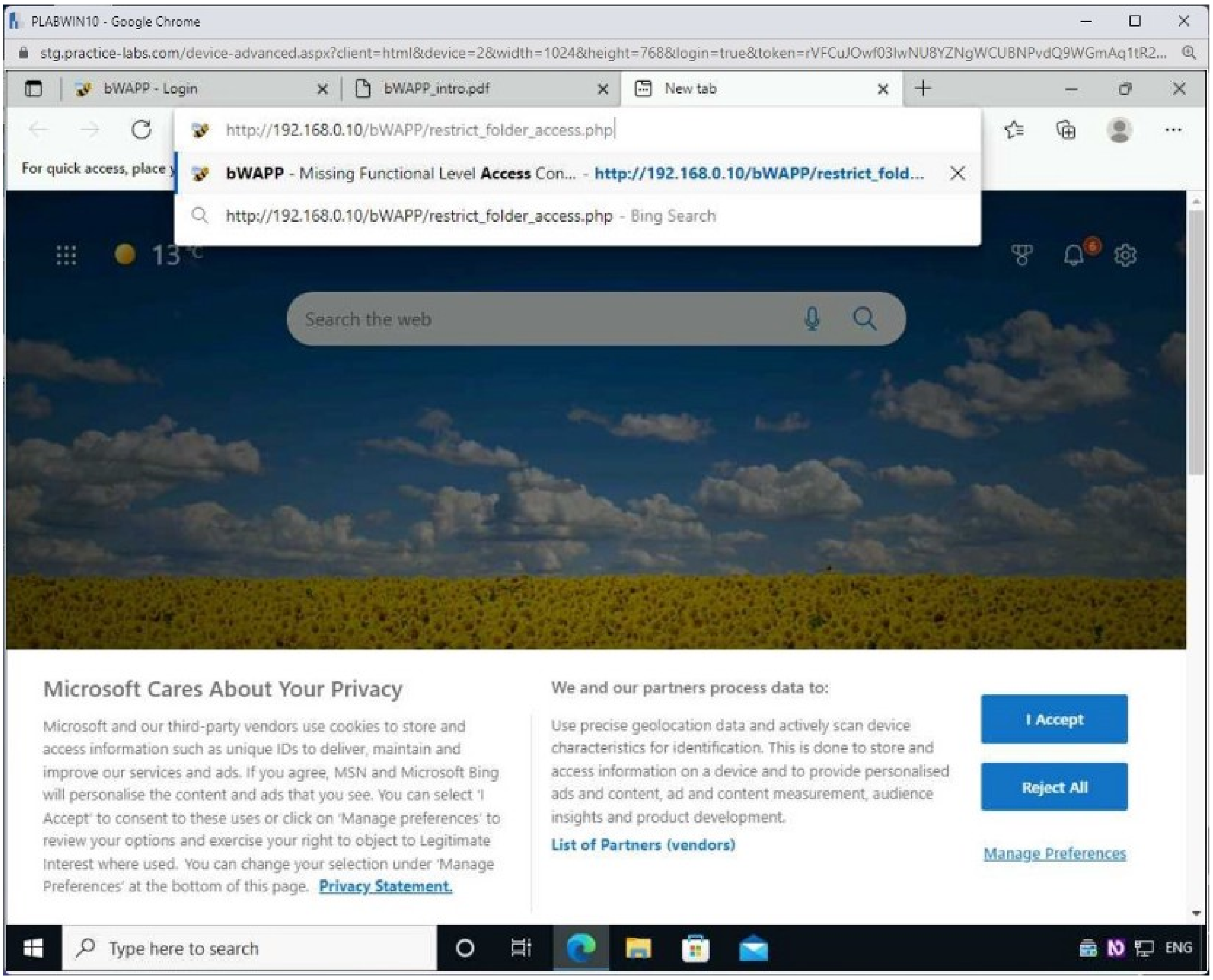
bWAPP is licensed under © 2014 MME BVBA / Follow @MME_IT on Twitter and ask for our cheat sheet, containing all solutions! / Ne

Step 12

In the address bar, type the following path:

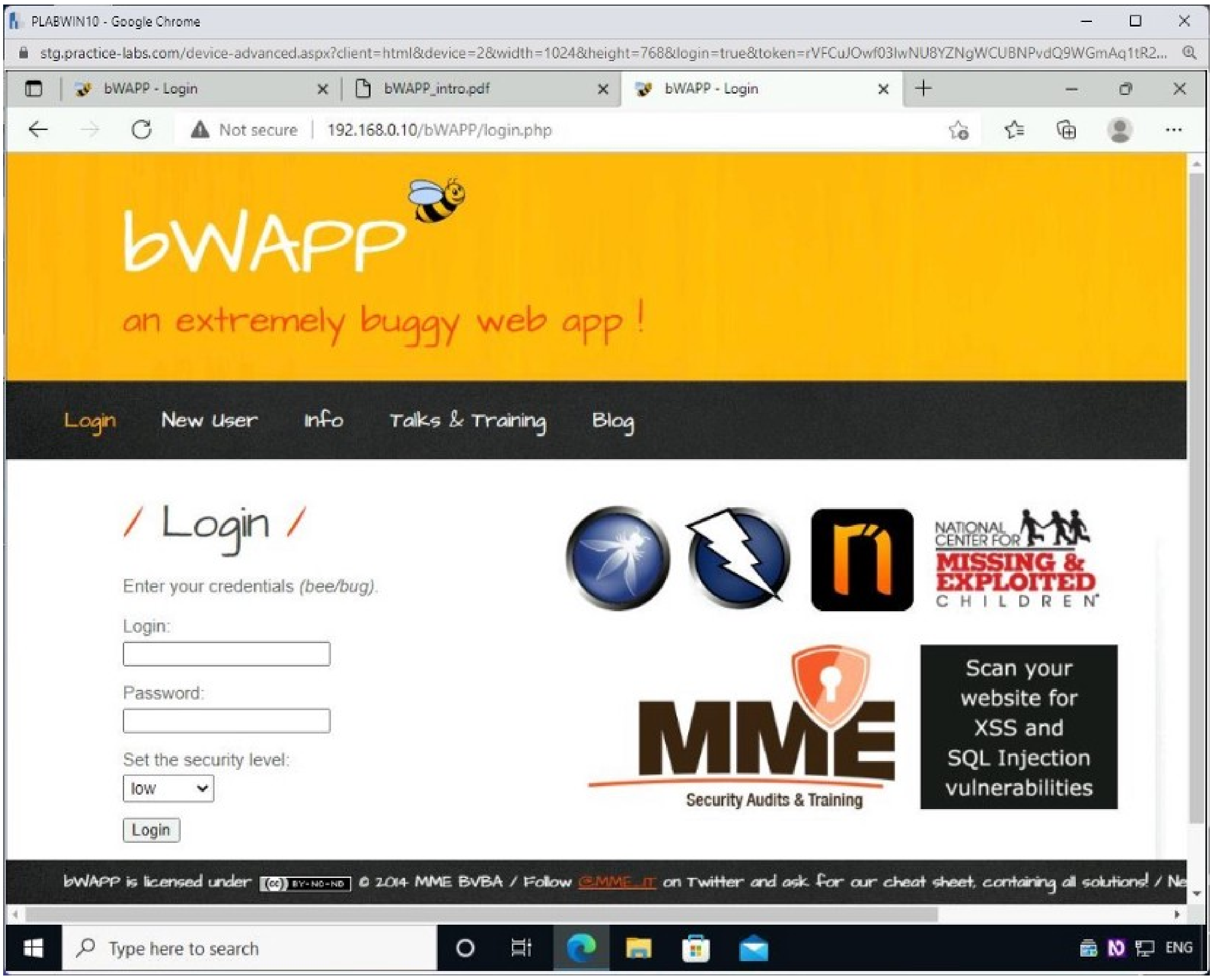
```
http://192.168.0.10/bWAPP/restrict_folder_access.php
```

Press **Enter**.


PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03IwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Login
bWAPP_intro.pdf
New tab
http://192.168.0.10/bWAPP/restrict_folder_access.php
bWAPP - Missing Functional Level Access Con... - http://192.168.0.10/bWAPP/restrict_fold...
http://192.168.0.10/bWAPP/restrict_folder_access.php - Bing Search
For quick access, place y
13 °C
Search the web
Microsoft Cares About Your Privacy
We and our partners process data to:
I Accept
Reject All
Manage Preferences
List of Partners (vendors)
Privacy Statement.
Type here to search
ENG

Step 13

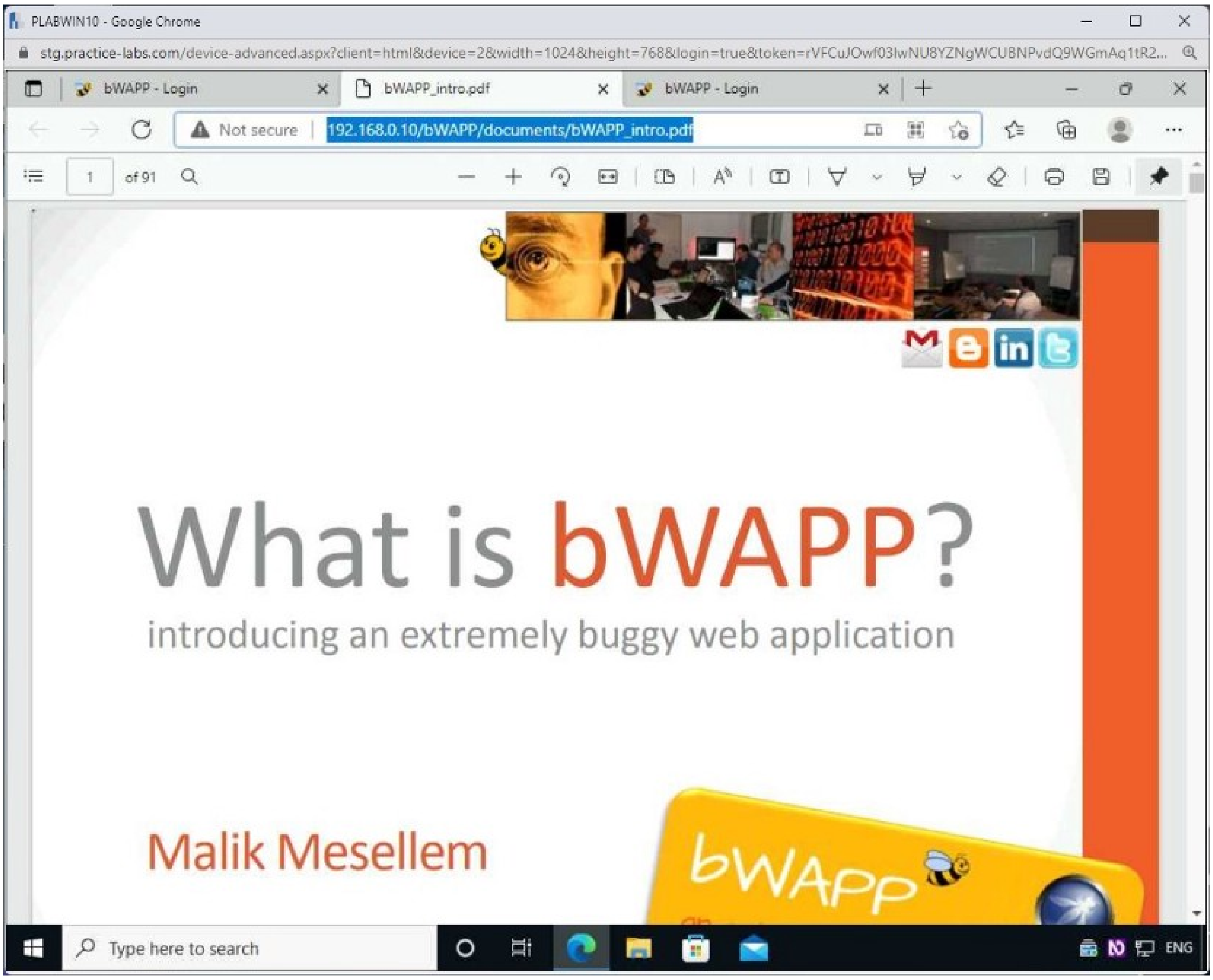
You are redirected to the login page, confirming that you have successfully logged out.


PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03IwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Login
bWAPP_intro.pdf
bWAPP - Login
Not secure
192.168.0.10/bWAPP/login.php
bWAPP
an extremely buggy web app !
Login
New User
Info
Talks & Training
Blog
/ Login /
Enter your credentials (bee/bug).
Login:
Password:
Set the security level:
low
Login
MME
Security Audits & Training
Scan your website for XSS and SQL Injection vulnerabilities
bWAPP is licensed under © 2014 MME BVBA / Follow @MME_IT on Twitter and ask for our cheat sheet, containing all solutions! / Ne
Type here to search
ENG

Step 14

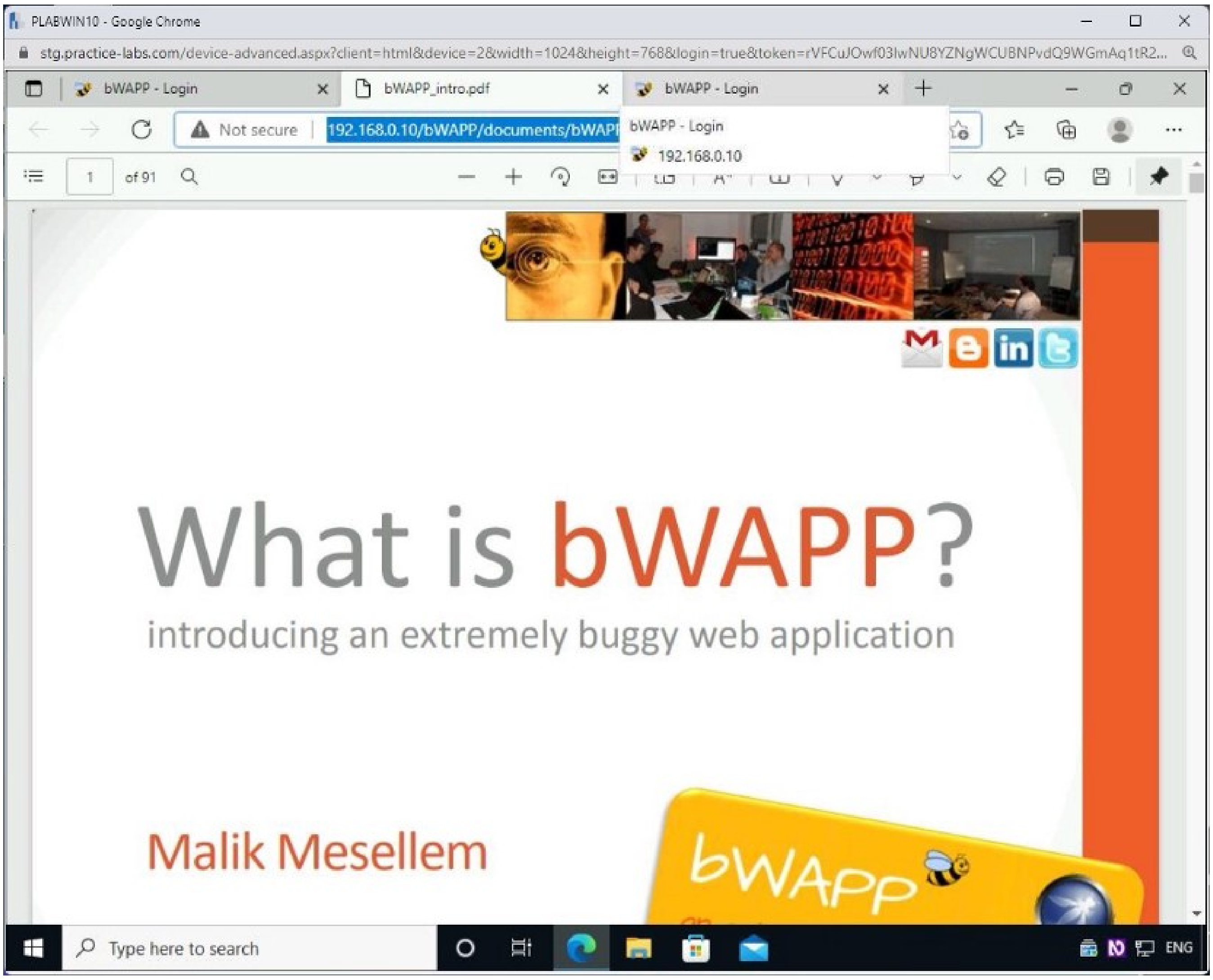
However, you will use the complete path to access files.

Go back to the second tab on which you had opened the PDF and copy the complete path.


PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03IwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Login
bWAPP_intro.pdf
bWAPP - Login
Not secure
192.168.0.10/bWAPP/documents/bWAPP_intro.pdf
1
of 91
What is bWAPP?
introducing an extremely buggy web application
Malik Mesellem
bWAPP
Type here to search
ENG

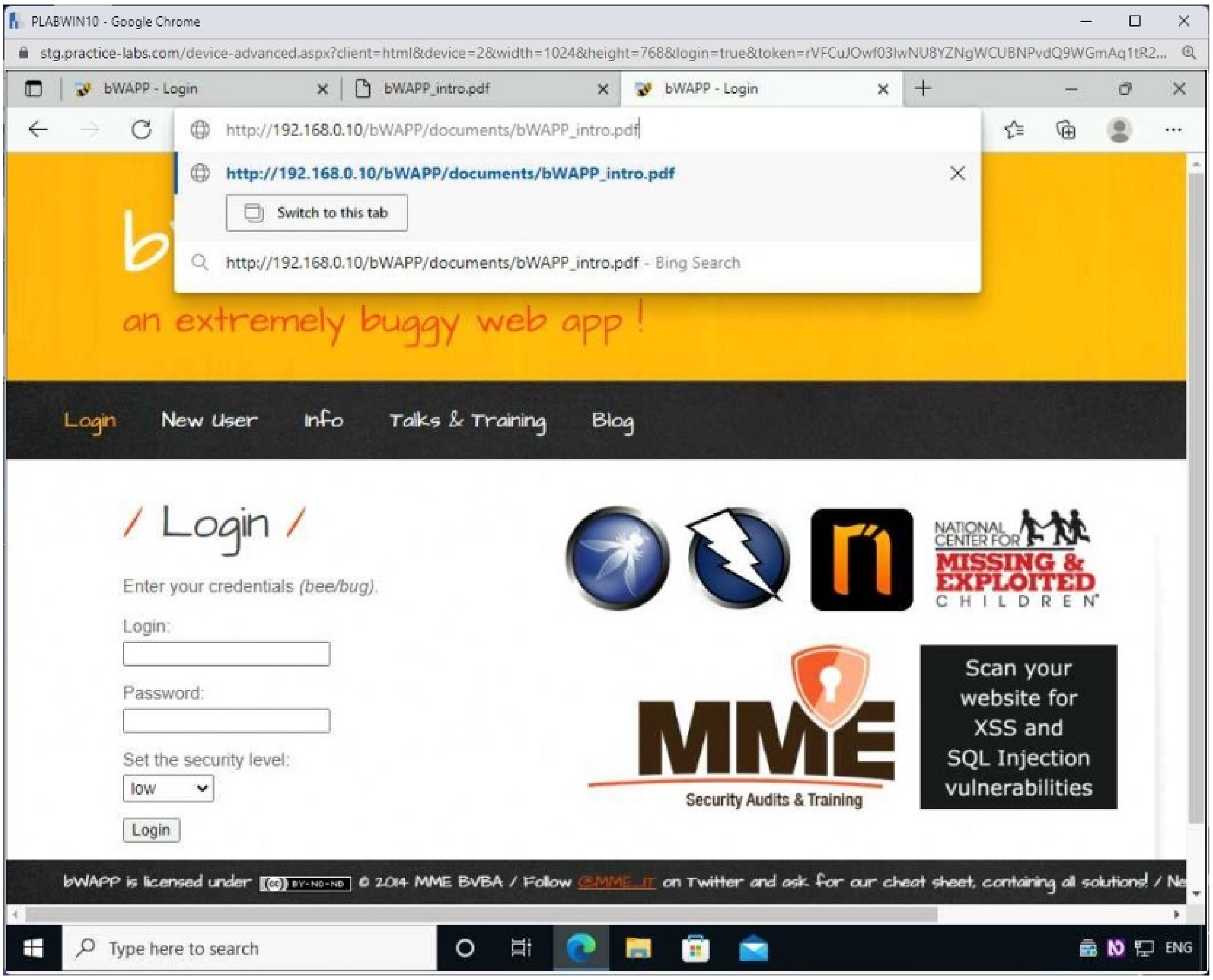
Step 15

Click the third tab — **bWAPP — Login**.


PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03lwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Login
bWAPP_intro.pdf
bWAPP - Login
Not secure
192.168.0.10/bWAPP/documents/bWAPP
bWAPP - Login
192.168.0.10
1
of 91
What is bWAPP?
introducing an extremely buggy web application
Malik Mesellem
bWAPP
Type here to search
ENG

Step 16

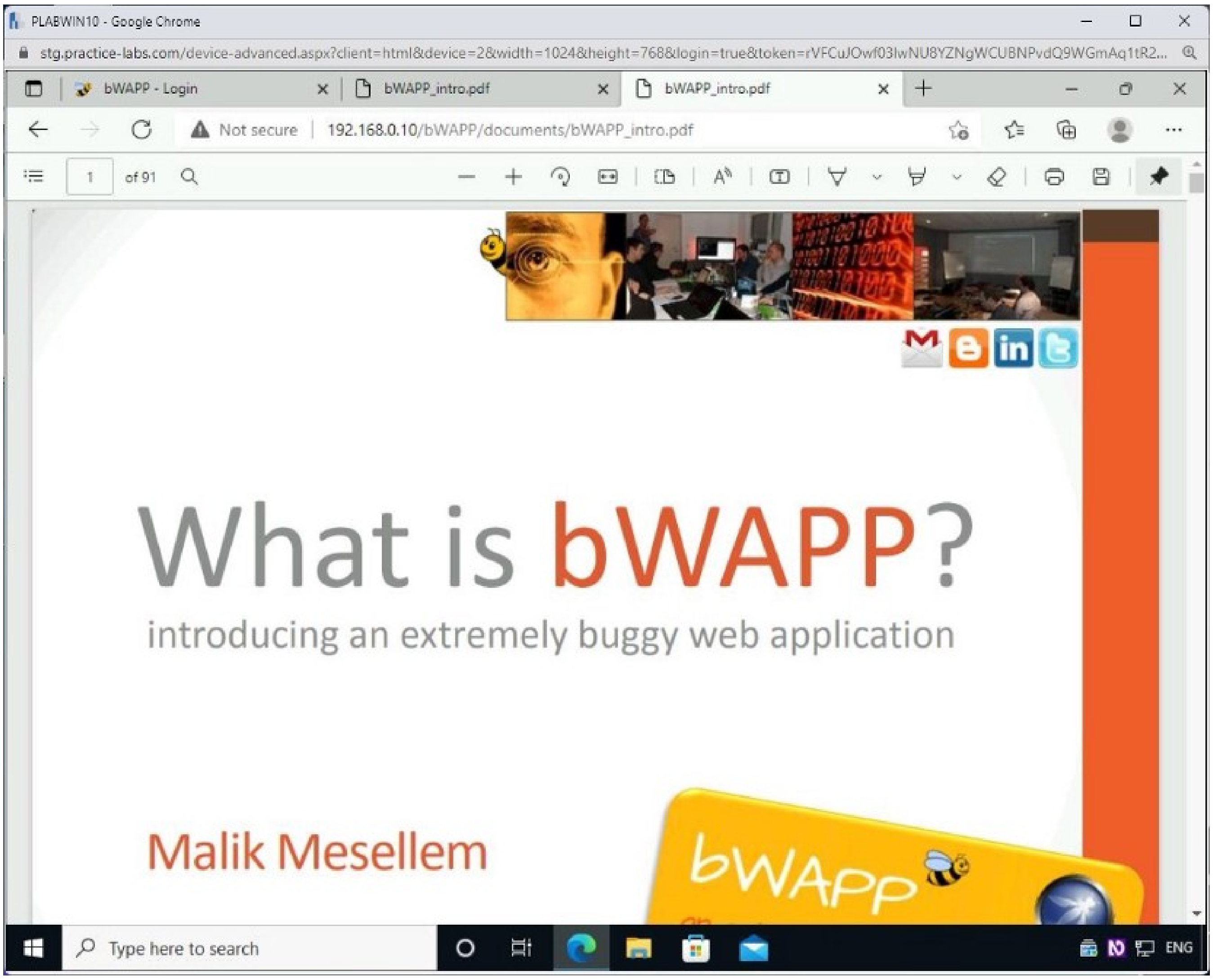
Overwrite the existing path with the one you copied from the second tab.


PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03IwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Login
bWAPP_intro.pdf
bWAPP - Login
http://192.168.0.10/bWAPP/documents/bWAPP_intro.pdf
http://192.168.0.10/bWAPP/documents/bWAPP_intro.pdf
Switch to this tab
http://192.168.0.10/bWAPP/documents/bWAPP_intro.pdf - Bing Search
an extremely buggy web app !
Login
New User
Info
Talks & Training
Blog
/ Login /
Enter your credentials (bee/bug).
Login:
Password:
Set the security level:
low
Login
Scan your website for XSS and SQL Injection vulnerabilities
MME Security Audits & Training
bWAPP is licensed under © 2014 MME BVBA / Follow @MME_IT on Twitter and ask for our cheat sheet, containing all solutions! / Ne
Type here to search
ENG

Step 17

Notice that the login page is no longer visible.

You are taken to the PDF file directly.


PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03IwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Login
bWAPP_intro.pdf
bWAPP_intro.pdf
Not secure
192.168.0.10/bWAPP/documents/bWAPP_intro.pdf
1
of 91
What is bWAPP?
introducing an extremely buggy web application
Malik Mesellem
bWAPP
Type here to search
ENG

Exercise 3 — Attack Access Controls

Access control attacks can be dangerous for a web application. They can be used to gain a higher level of access through privilege escalation, or gain administrative control.

An attacker uses a step-by-step approach to conduct an access control attack. At first, they identify users who have access to different data sets within an application and determines their roles and access. Then, they attempt to find an account with administrator functionality. Such functionalities are misconfigurations or vulnerabilities.

An attacker can exploit insecure access controls that are used within an application. these insecure access control methods include:

- Parameter-based Access Control

In this exercise, you will learn about parameter-based access control.

Learning Outcomes

After completing this exercise, you will be able to:

- Conduct Parameter-based Access Control Attack

Exercise 4 — Attack Session Management Mechanisms

You can view cookie information from unencrypted sites using the concept of session hijacking, which occurs at the network and application levels. At the application-level, an attacker can intercept the session ID of a particular session with the help of cookies and use it to gain unauthorized access to sensitive or critical data.

In this exercise, you will view the cookie and use it to launch another session.

Learning Outcomes

After completing this exercise, you will be able to:

- View Session ID and launch another session

ask 1 — View Session ID and Launch Another Session

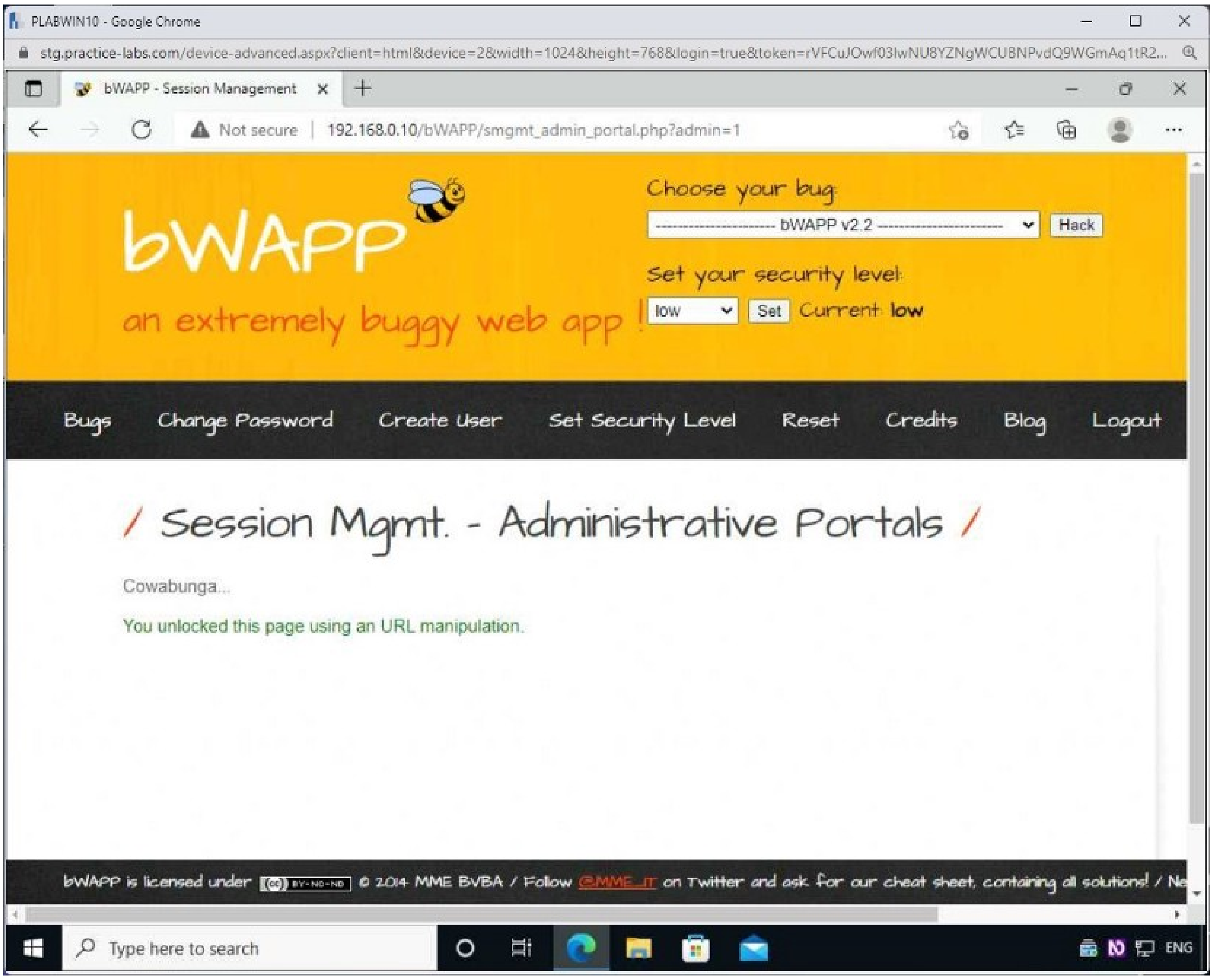
Session IDs are a critical part of a session established between a client and a server. They should never be revealed. If an attacker gets hold of a session ID, they can gain control of the overall session.

In this session, you will learn to view session ID and launch another session.

Step 1

Connect to **PLABWIN10**.

Ensure that the **Microsoft Edge** window is open. The **Session Mgmt — Administrative Portals** page is displayed.


PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03IwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Session Management
Not secure | 192.168.0.10/bWAPP/smgmt_admin_portal.php?admin=1
bWAPP
an extremely buggy web app !
Choose your bug
bWAPP v2.2
Hack
Set your security level
low
Set
Current: low
Bugs
Change Password
Create User
Set Security Level
Reset
Credits
Blog
Logout
Type here to search
ENG

/ Session Mgmt. - Administrative Portals /

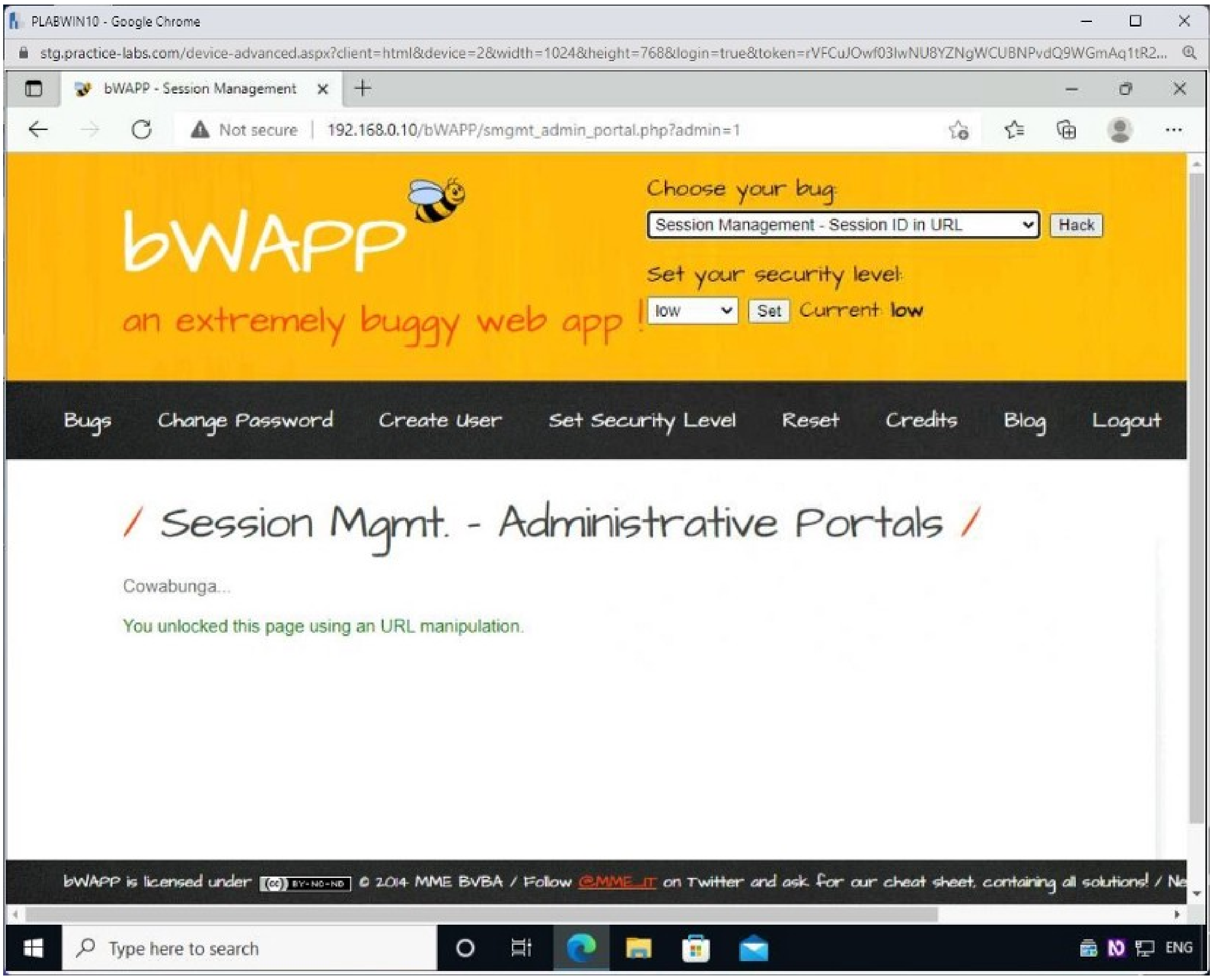
Cowabunga...

You unlocked this page using an URL manipulation.

bWAPP is licensed under (cc) BY-NC-ND © 2014 MME BVBA / Follow @MME_IT on Twitter and ask for our cheat sheet, containing all solutions! / Ne

Step 2

From the **Choose your bug** drop-down, select **Session Management — Session ID in URL** and click **Hack**.


PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03IwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Session Management
Not secure | 192.168.0.10/bWAPP/smgmt_admin_portal.php?admin=1
bWAPP
an extremely buggy web app !
Choose your bug:
Session Management - Session ID in URL
Hack
Set your security level:
low
Set
Current: low
Bugs
Change Password
Create User
Set Security Level
Reset
Credits
Blog
Logout
Type here to search
ENG

/ Session Mgmt. - Administrative Portals /

Cowabunga...

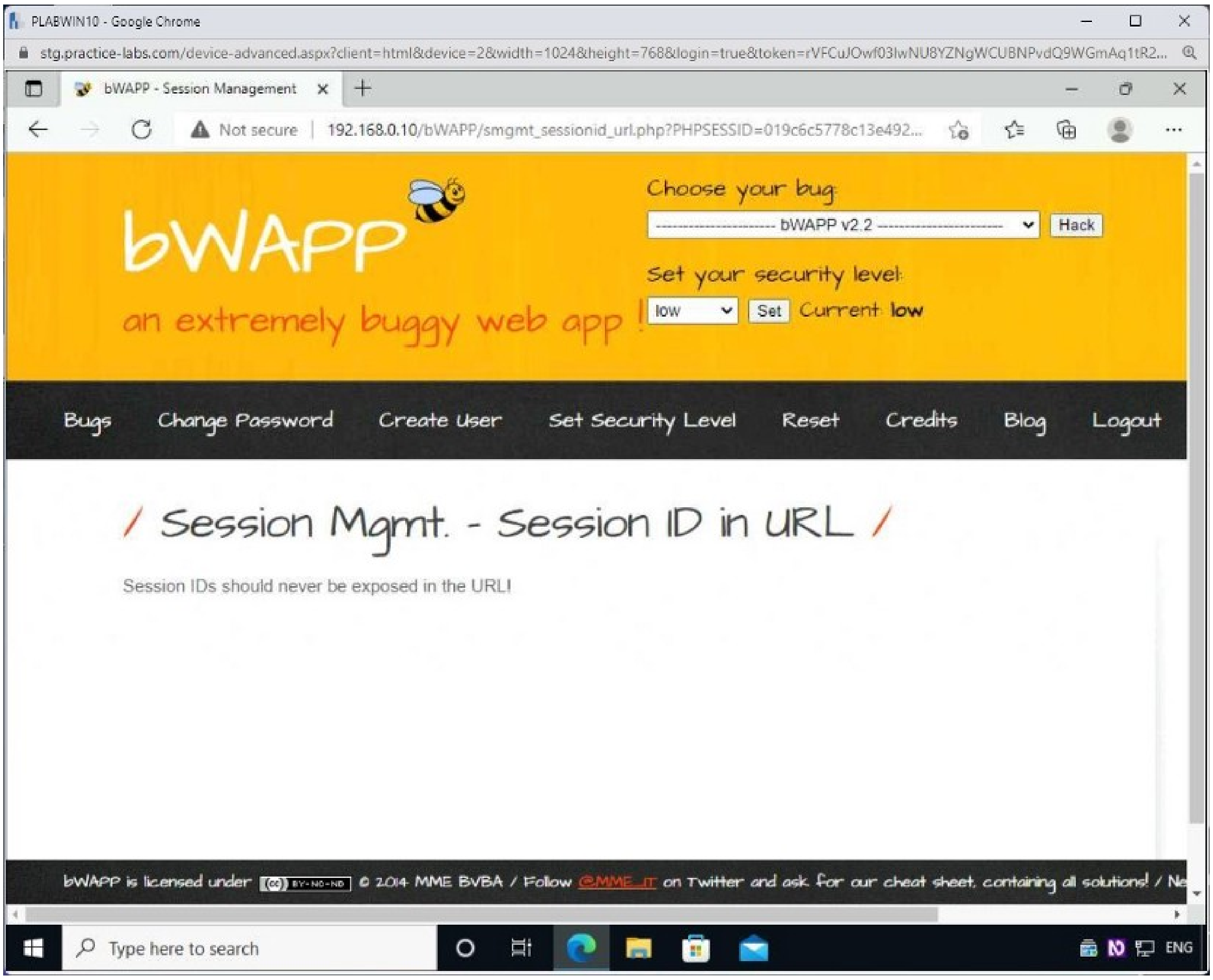
You unlocked this page using an URL manipulation.

bWAPP is licensed under BY-NC-ND © 2014 MME BVBA / Follow @MME_IT on Twitter and ask for our cheat sheet, containing all solutions! / Ne

Step 3

The **Session Mgmt. — Session ID** in the URL page is displayed.

Notice that the URL displays the **PHPSESSID** that contains the session ID.


PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03IwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Session Management
Not secure
192.168.0.10/bWAPP/smgmt_sessionid_url.php?PHPSESSID=019c6c5778c13e492...
bWAPP
an extremely buggy web app !
Choose your bug
bWAPP v2.2
Hack
Set your security level:
low
Set
Current: low
Bugs
Change Password
Create User
Set Security Level
Reset
Credits
Blog
Logout
bWAPP is licensed under © 2014 MME BVBA / Follow @MME_IT on Twitter and ask for our cheat sheet, containing all solutions! / Ne
Type here to search
ENG

/ Session Mgmt. - Session ID in URL /

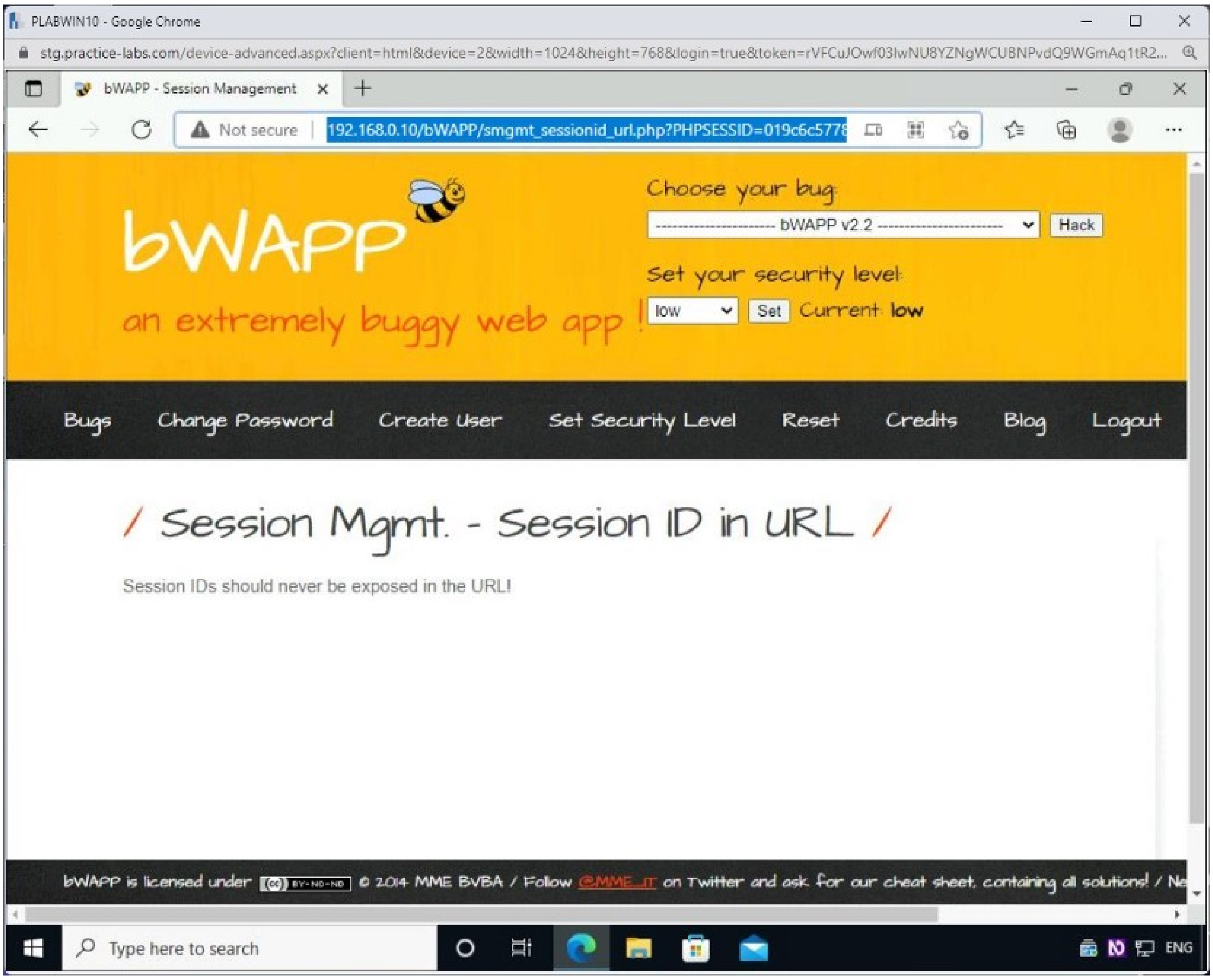
Session IDs should never be exposed in the URL!

Step 4

Typically, in most cases, if a web application is closed on one tab, the session should close.

However, because you know the session ID, you can simply copy the entire URL to launch it in another window.

Highlight the URL and copy it.


PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03IwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Session Management
Not secure
192.168.0.10/bWAPP/smgmt_sessionid_url.php?PHPSESSID=019c6c577
bWAPP
an extremely buggy web app !
Choose your bug
bWAPP v2.2
Hack
Set your security level:
low
Set
Current: low
Bugs
Change Password
Create User
Set Security Level
Reset
Credits
Blog
Logout
Type here to search
ENG

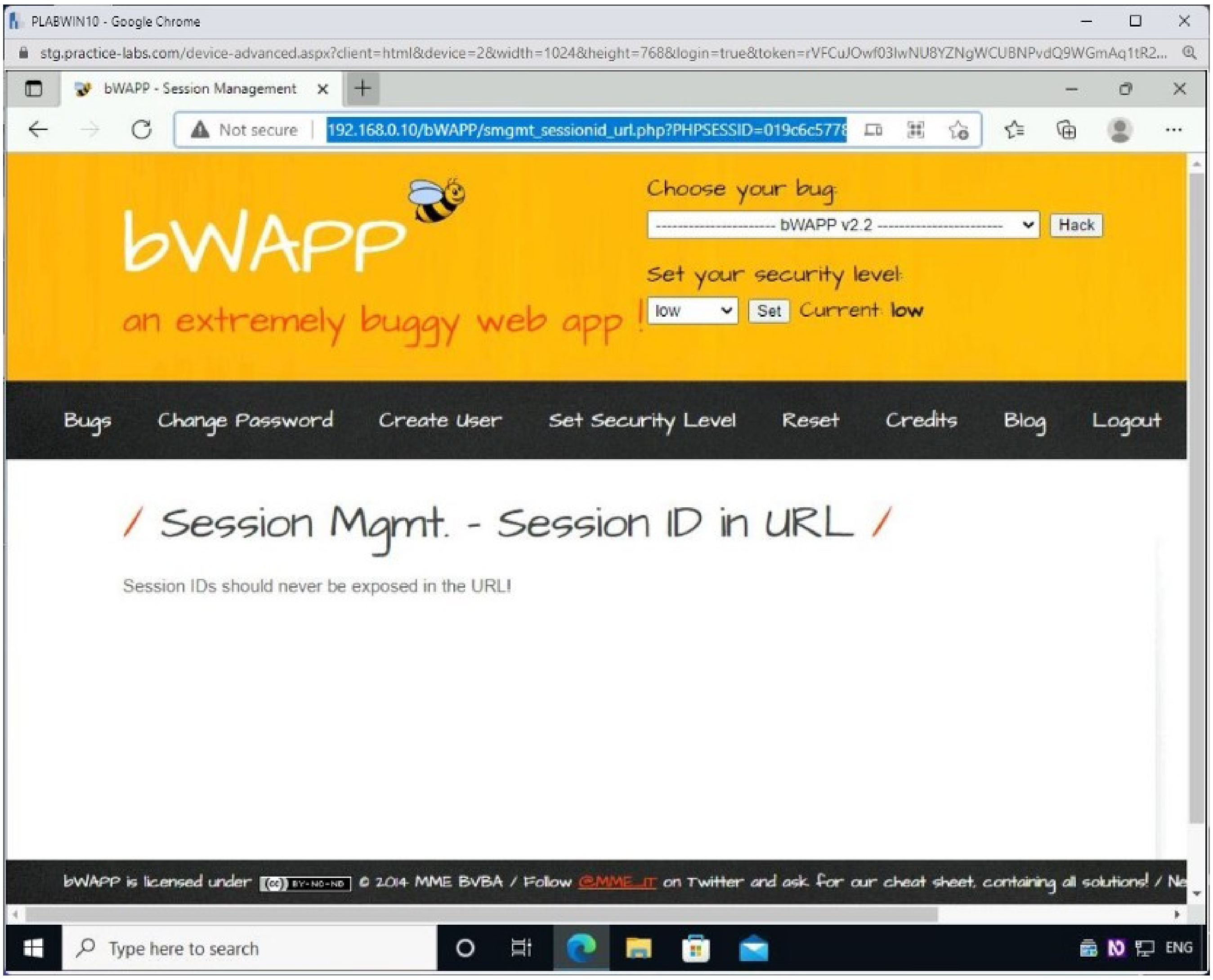
/ Session Mgmt. - Session ID in URL /

Session IDs should never be exposed in the URL!

bWAPP is licensed under BY-NC-ND © 2014 MME BVBA / Follow @MME_IT on Twitter and ask for our cheat sheet, containing all solutions! / Ne

Step 5

Open a new tab by clicking the + sign and then close the **bWAPP** tab.


PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03IwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Session Management
Not secure
192.168.0.10/bWAPP/smgmt_sessionid_url.php?PHPSESSID=019c6c577
bWAPP
an extremely buggy web app !
Choose your bug:
------------------- bWAPP v2.2 -------------------
Hack
Set your security level:
low
Set
Current: low
Bugs
Change Password
Create User
Set Security Level
Reset
Credits
Blog
Logout
bWAPP is licensed under © 2014 MME BVBA / Follow @MME_IT on Twitter and ask for our cheat sheet, containing all solutions! / Ne
Type here to search
ENG

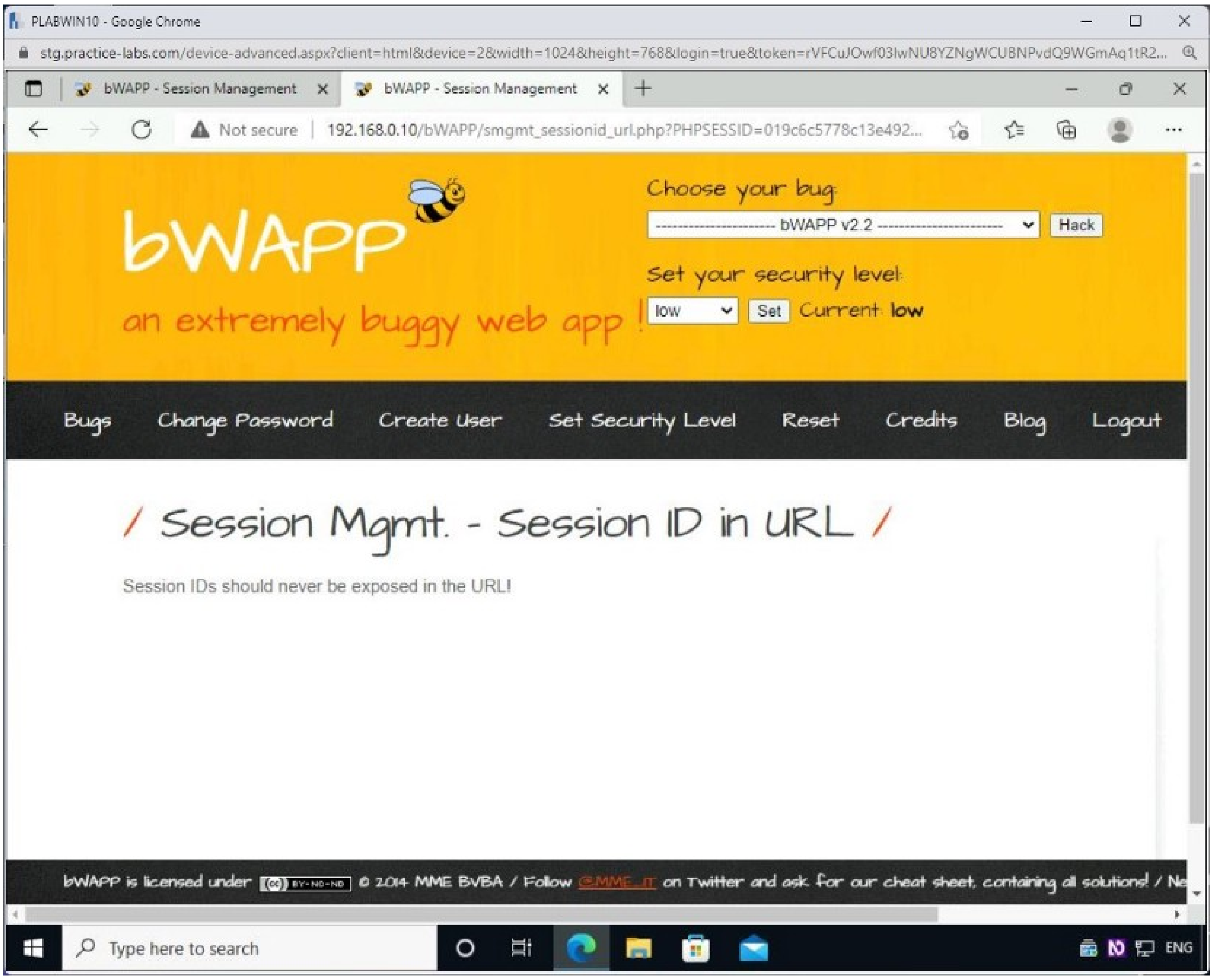
/ Session Mgmt. - Session ID in URL /

Session IDs should never be exposed in the URL!

Step 6

In the address bar, paste the new URL and press **Enter**.

The entire session will be reloaded.



PLABWIN10 - Google Chrome
stg.practice-labs.com/device-advanced.aspx?client=html&device=2&width=1024&height=768&login=true&token=rVFCuJOwf03IwNU8YZNgWCUBNPvdQ9WGmAq1tR2...
bWAPP - Session Management
bWAPP - Session Management
Not secure | 192.168.0.10/bWAPP/smgmt_sessionid_url.php?PHPSESSID=019c6c5778c13e492...
bWAPP
an extremely buggy web app !
Choose your bug
bWAPP v2.2
Hack
Set your security level:
low
Set
Current: low
Bugs
Change Password
Create User
Set Security Level
Reset
Credits
Blog
Logout

/ Session Mgmt. - Session ID in URL /

Session IDs should never be exposed in the URL!

bWAPP is licensed under (cc) BY-NC-ND © 2014 MME BVBA / Follow @MME_IT on Twitter and ask for our cheat sheet, containing all solutions! / Ne
Type here to search
ENG

Part 2

Learning Outcomes

In this module, you will complete the following exercises:

- Exercise 1 — Web Application Attack Methods
- Exercise 2 — Web API, Webhooks, and Web Shell
- Exercise 3 — Web Application Security

After completing this module, you will have further knowledge of:

- Attack Application Logic Flaws
- Attack Shared Environments
- Attack Database Connectivity

- Attack Web Services
- Web API
- Webhooks
- Web Shell
- Web Application Security Testing
- Source Code Review
- Web Application Encoding Schemes
- Web Application Countermeasures

Lab Duration

It will take approximately **30 minutes** to complete this lab.

Exercise 1 — Web Application Attack Methods

A web application can be attacked from different angles depending on its vulnerabilities. There can be vulnerabilities or flaws in the application logic, web services, or even the database client. The attackers look for these vulnerabilities and exploit the easy ones to exploit.

Learning Outcomes

After completing this exercise, you will have further knowledge of:

- Attack Application Logic Flaws
- Attack Shared Environments
- Attack Database Connectivity
- Attack Web Services

Attack Application Logic Flaws

Every single application is built with logic. Whichever functions are included in an application, each works with a specific login. For example, a function allows users to search for different documents within a repository. Even though this example is simple, logic defines how the search function works and provides relevant results.

One of the important points to note is that logic flaws are not easy to detect and cannot be detected by vulnerability scanners, and an attacker can attempt to find a logic flaw and exploit it. A close review of the source code by an experienced developer can prevent such issues.

Attack Shared Environments

Most organizations in the cloud environment use a shared infrastructure, which leads to several threats. For example, one of the biggest threats is the compromise of an application or data of one organization. If data is compromised for one customer, then other customers' data is also at stake.

Similarly, cloud service providers host the applications in a shared environment. If one application is compromised, other similar apps can be exploited for the same vulnerabilities. Administrative interfaces are mostly targeted with these attacks.

To prevent such attacks, you need to ensure the implementation of a strict access control mechanism. Also, a client should be separated with proper access implementation. Another prevention method is to ensure that hosts are properly patched, as this will reduce the attack surface for an attacker as there are fewer vulnerabilities.

Attack Database Connectivity

Applications connect to a database in the backend using a connection string. An attacker can change this connection string to connect it to a rogue database server, allowing control over a database. For example, an attacker can alter a connection string and switch off database encryption by removing a parameter.

There are various types of database connectivity attacks.

Some key examples include:

- **Hash Stealing**: This attack applies to Microsoft SQL Servers. When an application attempts to connect to a database, an attacker sniffs Windows credentials, which are in the form of hashes. An attacker then changes the DataSource and IntegratedSecurity parameters in the connection string with the name of their own Microsoft SQL server and sets the IntegratedSecurity parameter to No.
- **Port Scanning**: In this attack, an attacker would change the TargetPort parameter in the connection string and attempt to view generated messages. An attacker then keeps changing the port number to view different error messages.

- **Hijacking Web Credentials**: In this attack, an attacker connects directly with a database rather than an application via a web application system account. Once they have gained access, they modify the connection string and change the IntegratedSecurity parameter to True.

Attack Web Services

Web services provide functionality to web applications. Often, vulnerabilities within these web services lead to the exploitation of web applications. Once a web service is exploited, the underlying business logic and data are also compromised.

Attackers probe for the WSDL documents containing information about the service ports. Attackers use these documents to extract application information, such as message types and entry points into the application. An attacker creates valid SOAP requests containing malicious content and sends them to a web application. When errors are generated, an attacker can extract important information that can reveal vulnerabilities.

Exercise 2 — Web API, Webhooks, and Web Shell

Several applications integrate with other applications, and data can either be pulled or pushed using a web Application Programming Interface (API) or webhooks, out of which either can be used depending on specific requirements. This is similar to Web Shells, which are server-side programming-based exploit codes.

In this exercise, you will learn about Web API, Webhooks, and Web Shells.

Learning Outcomes

After completing this exercise, you will have further knowledge of:

- Web API
- Webhooks
- Web Shell

Web API

A Web API is an online service that works with the client applications and pulls or pushes the data from applications, and in most cases, Web APIs use the HTTP protocol to fetch and update information. You can create a Web API using various technologies, such as Java and .Net.

ClientAPIWeb Server & DatabaseRequestResponse

Figure 1.1: Diagram showing a Web API sitting between responses and requests to a web server & database.

Consider an example where you want to pull information from a Twitter account. You can design a Web API that will integrate into your application and pull the required information directly from Twitter.

There are different types of Web APIs that you can use depending on your needs.

- **SOAP**: is a web-based API that can enable communication between applications, even running on different operating systems, such as Windows and Linux. It uses HTTP and various other protocols, such as SMTP and FTP. SOAP itself is a protocol that defines the standards for API and allows data to be in XML format only. SOAP also does not define its own security.
- **REST**: REST is not a protocol or API but instead an architecture that can use SOAP or HTTP protocols. REST supports data in different formats, such as HTML, JSON, XML, and plain text. REST does not define its security like SOAP; instead, it uses the security of the method used for transporting the information.
- **RESTful**: The RESTful API is defined based on the REST architecture and uses HTTP as the protocol. The differences from REST include support for various HTTP methods, such as PUT, DELETE, GET, and GET. It has various features, such as stateless, cacheable, layered system, and client/server architecture.
- **XML-RPC**: XML-RPC is a protocol that uses XML to transfer data — same as SOAP. However, it is faster and consumes less bandwidth than the REST API.
- **JSON-RPC**: JSON-RPC is similar to XML-RPC, but it uses JSON-formatted data.

Webhooks

Webhooks are known as Push or Reverse APIs. They are intended to push information based on certain events. For example, when someone sends a friend request on Facebook, you get a notification or message on your mobile phone or email address.

ClientAPIWeb Server & DatabaseAPI RequestWebhook

Figure 1.2: Diagram showing a Webhook, sending data back to a client device after an API request.

Webhooks are event dependent in that if an event is not triggered, webhooks are not triggered either. With the immediate launch of a webhook, information is notified to a user in real-time. The important point to note about webhooks is that it is one-way communication. Another important point is the connectivity between the sending and receiving systems.

Web Shell

An attacker uses a server-side programming language, such as PHP, Perl, and ASP, to create a web shell; malicious code is deployed on a server remotely." to " A Web Shell is a malicious piece of code created by server-side languages such as Perl, RUBY, PHP, or Python that are installed on a web server and enable attackers remote access or remote administration over the target system.

AttackerWeb Server & DatabaseWeb shellBackdoor

Figure 1.3: Diagram showing a Web Shell being utilized by an attacker alongside a backdoor to a webserver.

An attacker uses various methods to deploy the web shell on the target server, including the following:

- SQL Injection
- Remote File Inclusion (RFI)
- Local File Inclusion (LFI)
- Vulnerability exploitation in the admin console

For example, if there is a SQL vulnerability, an attacker uses SQL Injection and exploits it. They then deploy a web shell on the target server. Using the capabilities of the web shell, they perform privilege escalation to get control over the data. Once an attacker has gained administrative access, they can exploit the application in different ways, such as modifying or exfiltrating data.

Some of the commonly used shells by attackers include:

- WSO Php Webshell
- C99
- R57
- b374k

***Note**: As of the time of writing, all of these tools are available on GitHub.*

Exercise 3 — Web Application Security

Several attacks can be performed on a web application. To handle these attacks, you need to implement different countermeasures. For example, methods to countermeasure the cross-site scripting attacks can differ from SQL Injection attacks. Therefore, you need to carefully review the web application security and plan for different scenarios.

In this exercise, you will learn about various web application security methods.

Learning Outcomes

After completing this exercise, you will have further knowledge of:

- Web Application Security Testing
- Source Code Review
- Web Application Encoding Schemes
- Web Application Countermeasures

Web Application Security Testing

A web application must be thoroughly tested to detect vulnerabilities and security flaws. To do this, organizations usually employ various methods of security testing. These methods are:

- **Manual**: Manual testing is performed to detect business logic flaws and threats. A tester can use custom code to test pre-existing code. Some of the key tools for manual testing include JMeter, Loadrunner, and Selenium.

- **Automated**: Automatic testing is a method of testing code at every development stage, allowing developers to perform fixes before moving ahead. Testing methods are programmed to conduct tests without any manual intervention. Some of the key tools for automated tests include TestComplete, Katalon Studio, and TestSigma.
- **Static Applications Security Testing (SAST**): SAST is a white-box method of testing in which a tester has access to code. The tester reviews the code and verifies it to comply with code writing guidelines and standards. The tester can use various tools, such as Appknox, AttackFlow, and BugScout.
- **Dynamic Applications Security Testing (DAST**): DAST is known as a black-box method of testing that is performed on a running application, in which a tester detects security flaws in the interface, sessions, and code injections. Testers do not know the system architecture or the application they are testing in this method, and can use various tools, such as Appknox and Netsparker.

Source Code Review

A source code review can be manual or automated. In a manual source code review, the process is time-consuming and requires a lot of skill. You need to read line by line and ensure that syntaxes are correct. If the tester is experienced, quite a few vulnerabilities can be caught. However, the downside of the manual source code review is that vulnerabilities can be missed as it is human-driven. One tester may catch one vulnerability but can be missed by another tester.

An automated source code review is fast and can track hidden vulnerabilities that could have been missed in a manual source code review. However, the downside of automated source code review is that it depends on the reviewing application's rules and parameters to detect vulnerabilities. It can easily miss vulnerabilities that are not programmed for detection within the application. It also requires a skilled tester to set up the automated source code testing environment.

Web Application Encoding Schemes

In web applications, users tend to input a variety of data. They may enter unusual characters as the input, further leading to different vulnerabilities. For example, a developer may use the <> or ' ' characters. You need to be able to handle them properly. Various encoding schemes can be used to handle these characters. The key methods are:

- **URL Encoding**: Converts a URL into an ASCII format. It also replaces the ASCII characters with a % and two-digit code hexadecimal for the ASCII character. An example is %20 for blank space.
- **HTML Encoding**: Converts an unusual character, such as >, into a format integrated into the HTML code. The > character is converted to &gt;.
- **Unicode Encoding**: Is of two types. In the 16-bit Unicode Encoding, the Unicode character is replaced with %u suffixed with the character's hexadecimal value. The UTF-8 method uses a variable length where each byte is replaced with the hexadecimal value prefixed with %. An example is %c2a8.
- **Base64 Encoding**: Converts the binary characters into ASCII characters. It is mainly used with email attachments.
- **Hex Encoding**: Converts the text into hex characters. An example of text PLAB is converted to 504c4142.

Web Application Attack Countermeasures

Different types of vulnerabilities require different countermeasures. To secure web applications, several countermeasures need to be applied, such as the following:

SQL Injection

An attacker can use SQL queries to access a backend database. An attacker uses an SQL Injection vulnerability within a web application and adds a malicious SQL script that allows them to gain access to a database. To prevent SQL Injection attacks, you can use some of the following countermeasures:

- Use Web Application Firewall (WAF)
- Use IDS for monitoring web application traffic
- Define the length of input that a user can perform
- Keep a web server and database server on separate systems and preferably on different segments — a web server in the DMZ and database server on a separate segment on an internal network
- Assign minimal permission to a service account on a database server

- Use parameterized queries and stored procedures to avoid dynamic SQL queries from users
- Use character filtering
- Validate and sanitize user input
- Safelist the user input
- Avoid calling of operating system functions and use built-in functions

Broken Authentication and Session Management

Several applications suffer from broken authentication and poor session management flaws, allowing attackers to access session information, secret keys, and passwords. Some key countermeasures include:

- Using TLS for transmitting data to and from the application
- Hash and salt user credentials
- Apply account lockout
- Enable multi-factor authentication
- Log authentication failures
- Use long and complex passwords
- Use random session IDs

Sensitive Data Exposure

Sensitive data can include various types of information, such as social security and credit card numbers. Any application that is making use of sensitive information needs to ensure that it is protected and secured by using key countermeasures, such as:

- AES encryption for data at rest
- TLS with HTTP Strict Transport Security (HSTS) for traffic in motion
- Always keep data encrypted with strong encryption algorithms
- Use data truncation or tokenization on the data that is not in use

- Keep encryption keys in secure offline storage
- Classify data and use it based on the classification
- Disable caching for sensitive data

Broken Access Control

In a broken access control attack, an attacker can access resources that should not be accessed. Such an attack leads to information disclosure, unwanted modification, or data destruction. To prevent broken access control attacks, you can use some of the key countermeasures:

- Perform access validation before granting access to a resource
- Configure session timeouts
- Use access permissions on files and directories
- Disable client-side caching of resources
- Configure removal of session tokens on the server

Security Misconfigurations

Sometimes it is not application vulnerabilities, but instead misconfigurations by administrators that get exploited. Some key countermeasures that can be implemented to handle security misconfigurations include:

- Disable all unnecessary services
- Close ports that are not required
- Change default credentials
- Perform vulnerability scans to discover vulnerabilities
- Use Secure flags on cookies with sensitive information
- Remove expired certificates and use a certificate from a trusted CA
- Avoid using HTTP and redirect all requests to HTTPS
- Encrypt all data at rest and in transmission

- Use complex passwords
- Remove all unnecessary features in an application or operating system
- Use strong access control

Cross-site Scripting (XSS) Attacks

An attacker embeds a malicious script into a web application in an XSS attack. To prevent XSS attacks, you should use some of the following key countermeasures:

- Validate forms and hidden fields for proper input
- Test web applications thoroughly and evaluate code using manual and automated testing
- Use fuzzers to test the input fields
- Use PKI to authenticate scripts being used
- Use a Web Application Firewall (WAF)
- Convert any character, other than alphanumeric, to HTML characters
- Define what is allowed rather than what is blocked
- Limit the size of the input fields
- Use Content Security Policy (CSP)

Insufficient Logging and Monitoring

You need to monitor your web application continuously. You also need to track the events taking place with the application. In several cases, administrators do not do this and realize its importance only after an attack has taken place. You should use the following key countermeasures in logging and monitoring:

- Create a baseline and monitor performance of web applications against it
- Enable user-specific logging to ensure that events can be traced back to users
- Define the scope for monitoring and logging — you do not need to log the successful logins, but failed ones should be tracked

- Ensure all failed events, such as failed logins, are logged
- Use centralized logging methods, such as Security Information and Event Management (SIEM), to analyze threats from events
- Log all critical events, such as high-value transactions

Directory Traversal

Using a directory traversal attack, an attacker can gain access to directories on a web server. An attacker may also gain access outside the root directory in this attack. To prevent this attack, you can use the following key countermeasures:

- Limit permissions on files and directories on a web server
- Use appropriate access permissions on restricted sections, such as the admin console, of a web application
- Patch up all vulnerabilities on web applications, web servers, and related technologies, such as Apache Web Server
- Validate the user input strings — use whitelisting to allow specific characters
- Perform filename sanitization

Cross-site Request Forgery (CSRF)

To prevent CSRF attacks, you can implement some of the following key countermeasures:

- Avoid saving login details in a web browser
- Log out properly from web applications
- Clear web browser history regularly
- Use a referrer header

Cookie Poisoning

Cookies can be used for various purposes, such as maintaining sessions or containing sensitive information, such as passwords. An attacker can modify the cookie's contents and gain access to the session. To prevent such attacks, you can use some of the following key countermeasures:

- Use strong encryption to encrypt passwords in a cookie
- Set a timeout on cookies to expire
- Clear cookies from web browsers regularly
- Use anti-malware applications to scan for malicious scripts, which hunt for cookies
- Ensure that authentication credentials in a cookie are tied to an IP address
- Verify content structure within a cookie
- Use a VPN to connect with the remote systems securely

Password Reset Attacks

Attackers exploit the improperly implemented password reset methods. For example, an application does not set the timeout on the password reset URLs. If this URL falls into an attacker's hands, they can reset the password. Several methods can be employed to prevent password reset attacks, such as:

- Set a timeout on the password reset URLs
- Use CAPTCHA with password reset requests
- Limit the number of requests from an IP address within a period
- Use multi-factor authentication (MFA)