Activité 9: Sécurisation des applications Web

Propriétés	Description
Intitulé long	Exploitation d'une plateforme d'apprentissage des vulnérabilités des applications <i>Web</i>
Intitulé court	Sécurisation des applications <i>Web</i>
Formation concernée	BTS Services Informatiques aux Organisations
Matière	Bloc 3 : Cybersécurité des services informatiques en deuxième année SLAM
Présentation	Ce Côté labo a pour objectif d'exploiter la plateforme d'apprentissage Portswigger.net du groupe OWASP (OpenWeb Application Security Project) afin de se familiariser avec les principales vulnérabilités des applications Web. Chaque activité couvre une problématique spécifique (SQLi, XSS, CSRF) en référence au top 10 des vulnérabilités décrites par l'OWASP. Dans un premier temps, l'étudiant doit comprendre le mécanisme des attaques. Dans un deuxième temps, l'objectif est de réaliser des défis à travers des manipulations pratiques. Cette neuvième activité concerne les problématiques liées à l'identification et l'authentification sur une application web. Cette vulnérabilité est classée n°7
	dans la classement OWASP 2021.
Compétences	 Protéger les données à caractère personnel ; Identifier les risques liés à la collecte, au traitement, au stockage et à la diffusion de données à caractère personnel. Garantir la disponibilité, l'intégrité et la confidentialité des services informatiques et des données de l'organisation face à des cyberattaques. Caractériser les risques liés à l'utilisation malveillante d'un service informatique ; Recenser les conséquences d'une perte de disponibilité, d'intégrité ou de confidentialité.
Savoirs	 Sécurité des applications web : risques, menaces et protocoles.
Prérequis	Administration d'un système <i>Linux.</i>
Outils	Une machine Kali Linux disposant d'un accès à internet et du logiciel BurpSuite (disponible sous Windows). Sites officiels : https://www.owasp.org.et.https://portswigger.net/burp/communitydownload
Mots-clés	OWASP, vulnérabilités, identification, authentification, BurpSuite, sniper.
Durée	Deux heures
Auteur(es)	Patrice Dignan, avec la relecture, les tests et les suggestions de Hervé Le Guern.
Version	v 1.0
Date de publication	01/05/25

Introduction		
IIPrésentation du type	de vulnérabilité	2
IIIExemples de codes v	/ulnérables	2
http://www.reseaucerta.org	©090 novembre 2025 – v1.0	Page 1/15

IVLes différentes formes d'authentification	.3
VContre-mesures de limitation et d'évitement	.3
VIAuthentification et réglementation	.3
VIIPrésentation des défis.	.3
VIIIRéalisation des défis	.4

I Introduction

L'identification et l'authentification sont des étapes clés de la sécurité dans une application web. L'identification consiste à déclarer qui l'on est (souvent avec un identifiant ou une adresse mail), tandis que l'authentification permet de prouver cette identité (par exemple en donnant un mot de passe).

Dans une application bien conçue, seules les personnes autorisées doivent accéder à certaines fonctionnalités ou données. Si l'identification ou l'authentification est mal gérée, des utilisateurs malveillants peuvent se faire passer pour d'autres.

C'est pourquoi il est essentiel, lors du développement, d'appliquer les bonnes pratiques de sécurité pour ces mécanismes.

II Présentation du type de vulnérabilité

Dans le classement OWASP Top 10 – 2021, la vulnérabilité "Identification and Authentication Failures" se classe en 7e position. Elle regroupe les failles qui permettent à un attaquant d'outrepasser l'authentification, de forcer des mots de passe ou encore d'accéder à des comptes sans autorisation.

Quelques exemples fréquents :

- CWE-287 : Authentification manquante ou incorrecte.
- CWE-798 : Utilisation de mots de passe codés en dur dans le code.
- CVE-2020-14750 : Faille dans Oracle WebLogic permettant une connexion sans authentification.

Conséquences : sur un site e-commerce, cela peut permettre de consulter des commandes d'autres clients, de modifier des données sensibles ou de détourner des paiements.

III Exemples de codes vulnérables

Quelques exemples de codes vulnérables :

PHP – Authentification simple et vulnérable :

```
if ($_POST['password'] == "admin123") {
    echo "Accès autorisé";
}
```

Mauvaise pratique : mot de passe en clair dans le code, sans gestion d'utilisateurs ni vérification sécurisée.

JavaScript (client-side) - Authentification côté client :

```
if (user === "admin" && password === "1234") {
    window.location = "admin.html";
}
```

L'authentification est visible et modifiable depuis le navigateur. Aucune sécurité côté serveur.

Python (Flask) – Session non protégée :

@app.route('/admin')
def admin():

```
if session.get('logged_in'):
return "Bienvenue admin"
```

Risque : si un utilisateur modifie la session sans vérification du rôle ou identifiant, il peut accéder à une page réservée.

IV Les différentes formes d'authentification

Il existe plusieurs formes d'authentification selon le facteur utilisé :

- 1. Connaissance : mot de passe, code PIN.
- 2. Possession : téléphone, carte à puce, clé USB sécurisée.
- 3. Inhérence : empreinte digitale, reconnaissance faciale.
- 4. Contexte : géolocalisation, horaire habituel, adresse IP.

L'authentification multifacteur (MFA) combine au moins deux de ces facteurs. Exemple : mot de passe + code envoyé par SMS.

L'authentification forte, elle, repose souvent sur des techniques cryptographiques, comme une signature numérique ou un certificat. Elle est plus difficile à contourner.

V Contre-mesures de limitation et d'évitement

Pour éviter les failles d'authentification, plusieurs bonnes pratiques sont recommandées :

- Ne jamais stocker de mots de passe en clair : utiliser des algorithmes comme bcrypt ou Argon2.
- Appliquer une politique de mots de passe forts.
- Protéger les pages critiques avec une authentification serveur, et gérer correctement les sessions.
- Implémenter une limite de tentatives (protection contre le brute force).
- Éviter les messages d'erreur trop explicites ("Mot de passe incorrect" ou plutôt "Utilisateur inexistant").
- Utiliser des bibliothèques de sécurité reconnues pour gérer l'authentification.

VI Authentification et réglementation

La CNIL et le RGPD imposent de sécuriser l'accès aux données personnelles. Cela concerne directement l'identification et l'authentification.

Selon la CNIL, l'accès à des données sensibles doit être protégé par une authentification forte.

Les développeurs doivent aussi :

- Prévoir un système de journalisation des accès.
- Informer les utilisateurs des traitements de leurs données.
- Permettre la modification et suppression des comptes.

Ne pas respecter ces règles peut entraîner des sanctions importantes (amendes, responsabilité juridique...).

VII Présentation des défis

Les défis permettant d'illustrer la problématique de l'authentification sont extrait du site portswigger.net. PortSwigger est un leader mondial dans la création d'outils logiciels pour les tests de

sécurité des applications Web. Ces défis nécessitent un accès à internet ainsi que le logiciel BurpSuite.

Avant de commencer les labos, il est nécessaire de créer un compte sur le site. La réalisation des défis nécessite l'utilisation du dossier do

Please enter your email address and password to log in.

Email address	patricedignan@gmail.com	
Password	••••••	•••••
	Forgot your password?	
	Remember me on this computer	
	Log in	Create account

Les trois défis proposés permettent d'explorer différentes failles liées à l'identification et à l'authentification :

- Défi n°1 : Énumération de logins et attaque par force brute L'objectif est d'identifier un identifiant valide, puis de deviner son mot de passe en testant de nombreuses combinaisons.
- Défi n°2 : Contournement d'une authentification à deux facteurs Ce défi consiste à contourner une authentification renforcée reposant sur l'envoi d'un code de vérification par e-mail.
- Défi n°3 : Usurpation de compte via la fonctionnalité "mot de passe oublié" Le but est d'exploiter une vulnérabilité dans le mécanisme de réinitialisation du mot de passe pour accéder frauduleusement à un compte utilisateur.

VIII Réalisation des défis

Le lien permettant d'accéder au labo est le suivant :

https://portswigger.net/web-security/authentication/password-based/lab-username-enumeration-viadifferent-responses

Ce laboratoire présente des vulnérabilités liées à l'énumération des noms d'utilisateurs et à la force brute des mots de passe. Un compte y est accessible à partir d'un identifiant et d'un mot de passe faciles à deviner, tous deux présents dans des listes de mots couramment utilisées (dictionnaires).

- Q1 : Effectuez les travaux préparatoires nécessaires pour le défi.
- Q2 : Suivez les étapes pour identifier un login valide.
- Q3 : Trouvez le mot de passe associé au login identifié pour valider le défi.

Défi n°2 : Contournement d'une authentification à double facteur

Le lien permettant d'accéder au labo est le suivant :

https://portswigger.net/web-security/authentication/multi-factor/lab-2fa-simple-bypass

L'authentification à deux facteurs de ce laboratoire peut être contournée. Vous disposez déjà d'un nom d'utilisateur et d'un mot de passe valides, mais vous n'avez pas accès au code de vérification 2FA de l'utilisateur. Pour résoudre le problème, accédez à la page du compte de Carlos.

Vos identifiants en tant qu'attaquant sont : wiener:peter Les identifiants de la victime sont : carlos:montoya

Q1 : Réalisez les préparatifs indispensables pour le défi.

Q2 : Mettez en œuvre l'attaque permettant de contourner la saisie d'un code de validation envoyé par email.

Q3 : Analysez ce qui rend cette vulnérabilité possible et indiquez les contre-mesures à adopter.

Défi n°3 : Réinitialisation malveillante d'un mot de passe

Le lien permettant d'accéder au labo est le suivant :

https://portswigger.net/web-security/learning-paths/authentication-vulnerabilities/vulnerabilities-inother-authentication-mechanisms/authentication/other-mechanisms/lab-password-reset-broken-logic

La fonctionnalité de réinitialisation du mot de passe de ce laboratoire est vulnérable. Pour résoudre le problème, réinitialiser le mot de passe de l'utilisateur Carlos, puis se connecter et accéder à sa page « mon compte ».

Vos identifiants en tant qu'attaquant : wiener/peter Login de la victime : carlos

Q1 : Effectuez les préparatifs nécessaires pour le défi.

Q2 : Validez le défi d'usurpation d'identité de Carlos en tirant parti de la vulnérabilité du formulaire de réinitialisation du mot de passe.

Q3 : Examinez les facteurs qui rendent cette vulnérabilité possible et proposez des contre-mesures appropriées.

Dossier documentaire

Défi n°1 : Énumération des utilisateurs via différentes réponses et force brute du mot de passe

Le lien permettant d'accéder au labo est le suivant :

https://portswigger.net/web-security/authentication/password-based/lab-username-enumeration-via-different-responses

Ce laboratoire présente des vulnérabilités liées à l'énumération des noms d'utilisateurs et à la force brute des mots de passe. Un compte y est accessible à partir d'un identifiant et d'un mot de passe faciles à deviner, tous deux présents dans des listes de mots couramment utilisées (dictionnaires).



Travaux préparatoires :

1- Depuis une machine Kali connectée à Internet, lancez **BurpSuite**. Configurez ensuite le navigateur pour qu'il utilise un **proxy local** : adresse **localhost** et **port 8080**. Dans BurpSuite, rendez-vous dans l'onglet **Proxy** et désactivez le mode **Intercept** (mettre sur *off*).

Burp	Project	Intruder	Repeater	View	Help				
Dash	board	Target	Proxy	Intr	uder	Repea	ter	Collaborator	Sequencer
Intere	cept	HTTP histor	y Web	Sockets	s history		Proxy	settings	
						-			
F	orward		Drop	Inte	ercept is of	ŧ j	Actio	on	Open browser

2- Téléchargez ensuite les deux fichiers dictionnaires contenant les **identifiants** et **mots de passe** à tester.

Pour cela, utilisez un éditeur de texte depuis la machine Kali afin de créer ou modifier ces fichiers.

Lab: Username enumeration via different responses



This lab is vulnerable to username enumeration and password brute-force attacks. It has an account with a predictable username and password, which can be found in the following wordlists:

- Candidate usernames
- Candidate passwords

To solve the lab, enumerate a valid username, brute-force this user's password, then access their account page.

Identification d'un login valide :

1- Sur la page d'accueil du défi, cliquez sur le lien « My account » situé en haut à droite de l'écran.

Home | My account



2- Dans BurpSuite, activez le **mode Proxy** (Intercept : *on*). Sur la page de connexion du laboratoire, entrez un **identifiant** et un **mot de passe incorrects**, puis cliquez sur le bouton **« Log In »** pour valider.

3- Une fois le bouton « Log In » cliqué, retournez dans BurpSuite et validez la requête interceptée en cliquant sur « Forward ». Ensuite, ouvrez l'onglet HTTP history, puis filtrez les requêtes affichées en recherchant la chaîne POST/login. Les détails de la requête envoyée et de la réponse reçue s'affichent dans la partie inférieure de la fenêtre.

Burp Pro	ject Intruder	Repeater	View Help											
Dashboard	d Target	Ргоху	Intruder	Repeater	Collaborator	Sequence	er Deco	der	Comparer	Logger	Organizer	Extensions	Learn	
Intercept	HTTP histo	ry We	ebSockets histor	/ 🗌 💮 Pr	oxy settings									
√ Filter	settings: Hiding C	SS, image ar	nd general binary	content; matchi	ng expression POST /l	ogin								
# Hort					Parame	Edited	Status sada	Longth	MIME type	Extension	Title	Notor	TIC	ID
# HUSL		Ivi	e V ORL		Parantis	Eulteu	Status code	Length	winvie type	Extension	Titte	Notes	113	1P
3277 https	://0aa300680475	9d728 PC	DST /login		1		200	3248	HTML		Username enume	eratio		79.125.84.16
3329 https	://0aa3006804/5	9d/28 PC	DST /login		~		200	3248	HTML		Username enum	eratio		79.125.84.16
3337 https	://0aa3006804/5	9d/28 PC	JST /login		~		200	3248	HIML		Username enum	eratio		79.125.84.16
33 https	://0aa3006804/5	9d728 PC	DST /login		~		200	3248	HTML		Username enum	eratio	~	79.125.84.16
3350 https	://0aa3006804/5	9d/28 PC	JSI /login				200	272024						/9.125.84.16
1636 https	://www.google.tr	G	EI /search?	=portswigger+i	ab+usern 🗸		200	3/3921	HIML		portswigger lab	usern	~	142.250.1/9.99
1665 https	s://portswigger.ne	et Gi	El /web-see	urity/authentica	tion/pass		200	45150	HIML		Lab: Username ei	nume	~	13.32.145.102
text/l	t. html,applicat:	ion/xhtml	+xml,applicat	ion/xml;q=0.	9,image/avif,imag	e/webp,*	/*;q=0	1 HTTP 2 Cont	2/2 200 0K ent-Type:	text/html	; charset=utf-8	3		
. 8								3 X-Fr	ame-Option	SAMEOR	IGIN	-		
6 Accept	t-Language: er	n-us,en;q tip defl:	=0.5					4 Cont	ent-Length	: 3140				
 Conter 	nt-Type: appl:	ication/x	-www-form-url	encoded				5						
9 Conte	nt-Length: 46		and total div					6 D0</td <td>OCTYPE html</td> <td>></td> <td></td> <td></td> <td></td> <td></td>	OCTYPE html	>				
10 Origin	n: https://0aa	30068047	59d728058940d	00b3000e. web	-security-academy	. net		7 <htm< td=""><td>nL></td><td></td><td></td><td></td><td></td><td></td></htm<>	nL>					
11 Refer	er: https://0a	aa3006804	759d728058940	c00b3000e. we	b-security-academ	y. net/lo	gin	8 5	liek brof	-	or (I obboodor (c	re (acadamul ab	Hoodon are n	
12 Upgra	de-Insecure-Re	equests: :	1					9	link href:	/resourc	es/res/labs res	s relectvlech	eet>	it-stytesneet/
13 Sec - Fe	etch-Dest: doo	ument						10	<title></title>	-/resourc	es/css/cabs.cs	s rec-scycesh	eeur	
14 Sec-F	.4 Sec-Fetch-Mode: navigate								Username	enumerat	ion via differe	ent responses		
15 Sec -Fe	etch-Site: sar	ne-origin												
16 Sec -Fe	etch-User: ?1							12 </td <td>/head></td> <td></td> <td></td> <td></td> <td></td> <td></td>	/head>					
17 Ie: t	railers							13 <	oody>					
19 usern	ame=invalid-ut	tilisateu	r&password=pa	ssword				14	<script sr<="" td=""></script>					

4- Dans la partie **Request** (à gauche), repérez la ligne contenant la valeur saisie pour le champ **username** (par exemple invalid-utilisateur).

Sélectionnez cette valeur avec la souris, faites un **clic droit**, puis choisissez **« Send to Intruder »** pour l'envoyer à l'outil Burp Intruder.

5- Rendez-vous dans l'onglet Intruder de BurpSuite.

Vous y verrez que la valeur du champ **username** est désormais encadrée par les symboles §, ce qui indique qu'elle sera utilisée comme **variable de test** lors de l'attaque.

0	Cho	Choose an attack type						
	Atta	ck type: Sniper						
0	Pay Conf	load positions figure the positions where payloads will be inserted, they can be added into the target as well as the base request.						
		Target: https://0aa3006804759d728058940c00b3000e.web-security-academy.net						
	6	Accept-Language: en-US en: (EN-S						
	7	Accept Encoding: gzip, deflate, br						
	8	Content-Type: application/x-www-form-urlencoded						
	9	Content-Length: 46						
	10	Origin: https://0aa3006804759d728058940c00b3000e.web-security-academy.net						
	11	Referer: https://0aa3006804759d728058940c00b3000e.web-security-academy.net/login						
	12	Upgrade_Insecure_Requests: I						
	13	SerFetch-Dest: document						
	15	Sec-Fetch-Site: same-origin						
	16	Sec-Fetch-User: 71						
	17	Te: trailers						
	18							
	19	username=§invalid-utilisateur§&password=password						

Vérifiez que le mode Sniper est bien activé avec l'option Simple list.

?	Payload sets	5		
	You can define	one or more paylo	oad sets. The n	umber of payload sets depends on the attack
	Payload set:	1	\sim	Payload count: 0
	Payload type:	Simple list	\sim	Request count: 0

6- Dans l'onglet **Payloads**, chargez la liste de logins créée précédemment. Une fois la liste ajoutée, cliquez sur « Start attack » pour lancer l'attaque.

Paste	carlos	
Lord	root	
Load	admin	
Remove	test	
itemore.	guest	
Clear	info	
	adm	
Deduplicate	mysql	
	user	
	administrator	

L'attaque prend un certains temps. Ne pas hésiter à faire une pause en prenant un café.

7 - Une fois l'attaque terminée, examinez la colonne **Length**, qui indique la longueur de la réponse reçue pour chaque tentative. Vous constaterez qu'un des identifiants testés génère une réponse plus longue que les autres, accompagnée d'un message différent. Alors que la majorité des requêtes

affichent « **Invalid username** », celle-ci retourne « **Incorrect password** ». Cela signifie que nous avons identifié un login valide. Il reste maintenant à trouver le mot de passe associé à ce compte.

Force brute du mot de passe du login valide :

1 - Fermez la fenêtre de l'attaque précédente, puis retournez dans l'onglet Positions de **Burp Intruder**. Cliquez sur le bouton « **Clear** » pour réinitialiser les marqueurs. Remplacez ensuite l'ancien identifiant par celui que vous avez identifié comme valide. Puis, sélectionnez avec la souris la valeur du champ mot de passe, et cliquez sur « **Add** » pour définir cette partie comme variable de test, en restant en mode Sniper.

username=identified-user&password=§invalid-password§

?	Payload positions Configure the positions where payloads will be inserted, they can be added into the target as well as the base request.			
	Target: https://0adc000103616ff78109f28b00da005e.web-security-academy.net	Update Host header to match	target	Add § Clear §
	© Accept-Language: en-US an;qr0.5 7 Accept-Encoding: gzip, deflate, br 8 Content-Type: application/x-www-form-urlencoded			Auto §
	Content-Length: 46 Origin: https://Dok/CO001085L6F78109f28b004a005s.web-security-academy.net Refere: https://Dok/CO001085L6F78109f28b004a005s.web-security-academy.net/login Upgrade-Drascure-Requests: 1 Sec-Fetch-Dest: document Sec-Fetch-Tost: are-origin Sec-Fetch-User: 11 Te: trailers Is uername=anahelm6passwords5password5			Refresh
	$\bigcirc \oslash (\widehat{\mathbf{e}}) [\widehat{\mathbf{e}}]$ [Search	<i>ب</i> م	1 highlight	Clear
	1 payload position	1	Length: 792	

2- Dans l'onglet **Payloads**, effacez la liste des identifiants précédemment utilisée. Remplacez-la par la liste des mots de passe que vous avez préparée, puis cliquez sur « **Start attack** » pour lancer l'attaque.

Payload settings [Simple list]

This payload type lets you configure a simple list of strings that are used as payloads.

Paste	123456	
	password	
Load	12345678	
Parraua	qwerty	
Remove	123456789	
Clear	12345	
	1234	
Deduplicate	111111	
	1234567	
	dragon	
	172172	
Add	Enter a new item	
Add from list [Tra version only]	

3- La ligne de réponse correspondant au code de retour 302 indique le bon mot de passe.

R	equest	Resp	onse						
P	retty	Raw	Hex	Render					
l	HTTP/2	302 Fou	und						
2	Location: /my-account?id=anaheim								
З	Set-Cookie: session=rsZTGmFAU8ttl4klJSfvqqFahH4tm0SH; Secure; HttpOnly; SameSite=None								
4	X-Frame-Options: SAMEORIGIN								
5	Content-Length: 0								

4- S'authentifier avec le login et le mot de passe trouvé pour valider le défi.



Défi n°2 : Contournement d'une authentification à double facteur

Le lien permettant d'accéder au labo est le suivant :

https://portswigger.net/web-security/authentication/multi-factor/lab-2fa-simple-bypass

L'authentification à deux facteurs de ce laboratoire peut être contournée. Vous disposez déjà d'un nom d'utilisateur et d'un mot de passe valides, mais vous n'avez pas accès au code de vérification 2FA de l'utilisateur. Pour résoudre le défi, il faut accéder à la page du compte de Carlos.

Vos identifiants en tant qu'attaquant sont : wiener:peter

Les identifiants de la victime sont : carlos:montoya





Home | My account



<u>Travaux préparatoires :</u>

1- Connectez-vous avec le compte attaquant dont l'identifiant est **wiener**, puis lancez BurpSuite en veillant à ce que le mode Intercept soit désactivé (**Intercept : off**).

Login		
Username		
wiener		
Password		
•••••		

2- Ouvrez le lien vers votre **boîte mail** (Email client) pour consulter le **code de vérification** envoyé après la saisie de votre mot de passe, dans le cadre de la **double authentification**.

Your email add	lress is wiener@exploit-0a22	200df0363beba800f6b6d014e0	Of6.explo	it-server.net	
Displaying all emails @exploi	t-0a2200df0363beba800f6b6d014e00f6.exploit-server.net	and all subdomains			
Sent	То	From	Subject	Body	
2025-04-21 14:01:57 +0000	wiener@exploit- 0a2200df0363beba800f6b6d014e00f6 .exploit-server.net	no- reply@0a9900bf038cbe8080eb6c2a003b 00ce.web-security-academy.net	Security code	Hello! Your security code is 0 156. Please enter this in th e app to continue. Thanks, Support team	View raw

3 - Copiez le code reçu par mail et utilisez-le pour finaliser l'authentification. En analysant la requête dans BurpSuite, on observe que la validation du code modifie la fin de l'URL, qui devient alors /my-account en cas de succès. Il peut donc être pertinent de tester manuellement une modification de l'URL, en remplaçant sa fin par /my-account, après avoir validé la première étape de l'authentification (identifiant et mot de passe) sur le compte de la victime.

4- Déconnectez-vous du compte wiener en vous déauthentifiant depuis l'interface du site.

Réalisation de l'attaque :

1 - Activez le mode Intercept dans BurpSuite (Intercept : on). Connectez-vous ensuite avec le compte de Carlos, puis cliquez sur le bouton **Forward** dans BurpSuite pour laisser passer la requête.

Intercept HTTP history WebSockets history Image: Proxy settings Image: Pretty Request to https://0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net:443 [79.125.84.16] Forward Drop Intercept is on Action Open browser Pretty Raw Hex 1 GET /login2 HTTP/2 2 Host: 0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net 3 Cookie: session=Eu9P000nV2IkrJtbquuCn60M0jF3wMkZ 4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0 5 Accept: text/html, application/xhtl+xml, application/xml; q=0.9, image/avif, image/webp, */*; q=0.8 6 Accept-Language: en-US, en; q=0.5 7 Accept-Encoding: gzip, deflate, br 8 Referer: https://0a9900bf038cb8080eb6c2a003b00ce.web-security-academy.net/login 9 Upgrade-Insecure-Requests: 1 Sec-Fetch-Dest: document 10 Sec-Fetch-Dist: same-origin 13 Sec-Fetch-User: ?1 Te: 14 Te: trailers	Da	shboard	Target	Proxy	Intruder	Repeater	Collaborator	Sequencer	Decoder	(
<pre>Request to https://0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net:443 [79.125.84.16] Forward Drop Intercept is on Action Open browser Pretty Raw Hex GET /login2 HTTP/2 Host: 0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net Cookie: session=Eu9PQ00nV21krJtbquuCn6QM0jF3wHkZ User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8 Accept-Encoding: gzip, deflate, br Referer: https://0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net/login Upgrade-Insecure-Requests: 1 Sec-Fetch-Dest: document Sec-Fetch-Site: same-origin Sec-Fetch-User: 71 Te: trailers </pre>	Int	ercept	HTTP histor	y Web	Sockets history	() i	Proxy settings			
ForwardDropIntercept is onActionOpen browserPrettyRawHex1GET /Login2 HTTP/22Host: 0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net3Cookie: session=Eu9PQ00nV2IkrJtbquuCn60M0jF3wMkZ4User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.05Accept: text/html.application/xhtml+xml.application/xml;q=0.9, image/avif, image/webp, */*;q=0.86Accept-Language: en-US, en;q=0.57Accept-Encoding: gzip, deflate, br8Referer: https://0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net/login9Upgrade-Insecure-Requests: 110Sec-Fetch-Dest: document11Sec-Fetch-Site: same-origin12Sec-Fetch-Site: same-origin13Sec-Fetch-User: ?114Te: trailers	0	A Request	to https://0a9	900bf038cb	e8080eb6c2a00	3b00ce.web-	security-academy.net	443 [79.125.84.16	5]	
Pretty Raw Hex 1 GET /login2 HTTP/2 2 Host: 0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net 3 Cookie: session=Eu9PQ00nV2IkrJtbquuCn6QM0jF3wMkZ 4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0 5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8 6 Accept-Language: en-US, en;q=0.5 7 Accept-Encoding: gzip, deflate, br 8 Referer: https://0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net/login 9 Upgrade-Insecure-Requests: 1 10 Sec-Fetch-Dest: document 11 Sec-Fetch-Dist: same-origin 12 Sec-Fetch-User: ?1 14 Te: trailers		Forward		Drop	Intercept is	on	Action Op	en browser		
<pre>1 GET /login2 HTTP/2 2 Host: 0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net 3 Cookie: session=Eu9PQ00nV2IkrJtbquuCn6QM0jF3wMkZ 4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0 5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8 6 Accept-Language: en-US,en;q=0.5 7 Accept-Encoding: gzip, deflate, br 8 Referer: https://0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net/login 9 Upgrade-Insecure-Requests: 1 10 Sec-Fetch-Dest: document 11 Sec-Fetch-Mode: navigate 12 Sec-Fetch-Site: same-origin 13 Sec-Fetch-User: ?1 14 Te: trailers 15</pre>	Pr	etty R	aw Hex							
<pre>Host: 0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net Cookie: session=Eu9P000nV2IkrJtbquuCn60M0jF3wMkZ User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8 Accept-Language: en-US,en;q=0.5 Accept-Encoding: gzip, deflate, br Referer: https://0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net/login Upgrade-Insecure-Requests: 1 Sec-Fetch-Dest: document Sec-Fetch-Mode: navigate Sec-Fetch-User: ?1 Te: trailers</pre>	1	GET /log:	in2 HTTP/2							
<pre>3 Cookie: session=Eu9PQ00nV2IkrJtbquuCn6QM0jF3wMkZ 4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0 5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8 6 Accept-Language: en-US,en;q=0.5 7 Accept-Encoding: gzip, deflate, br 8 Referer: https://0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net/login 9 Upgrade-Insecure-Requests: 1 10 Sec-Fetch-Dest: document 11 Sec-Fetch-Mode: navigate 12 Sec-Fetch-Site: same-origin 13 Sec-Fetch-User: ?1 14 Te: trailers 15</pre>	2	Host: 0a9	9900bf038cb	e8080eb6c2	2a003b00ce.w	eb-security	-academy.net			
<pre>4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0 5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8 6 Accept-Language: en-US,en;q=0.5 7 Accept-Encoding: gzip, deflate, br 8 Referer: https://0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net/login 9 Upgrade-Insecure-Requests: 1 10 Sec-Fetch-Dest: document 11 Sec-Fetch-Mode: navigate 12 Sec-Fetch-Site: same-origin 13 Sec-Fetch-User: ?1 14 Te: trailers</pre>	3	Cookie: s	session=Eu9	PQ00nV2Ikr	JtbquuCn6QM	0jF3wMkZ	-			
<pre>5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8 6 Accept-Language: en-US,en;q=0.5 7 Accept-Encoding: gzip, deflate, br 8 Referer: https://0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net/login 9 Upgrade-Insecure-Requests: 1 10 Sec-Fetch-Dest: document 11 Sec-Fetch-Dest: document 12 Sec-Fetch-Site: same-origin 13 Sec-Fetch-User: ?1 14 Te: trailers 15</pre>	4	User-Ager	nt: Mozilla	/5.0 (X11;	Linux x86_0	54; rv:109.	0) Gecko/2010010	l Firefox/115.	0	
<pre>6 Accept-Language: en-US,en;q=0.5 7 Accept-Encoding: gzip, deflate, br 8 Referer: https://0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net/login 9 Upgrade-Insecure-Requests: 1 10 Sec-Fetch-Dest: document 11 Sec-Fetch-Mode: navigate 12 Sec-Fetch-Site: same-origin 13 Sec-Fetch-User: ?1 14 Te: trailers 15</pre>	5	Accept: 1	text/html,a	pplication	n/xhtml+xml,	application	/xml;q=0.9,image	/avif,image/we	ebp,*/*;q=0	. 8
<pre>7 Accept-Encoding: gzip, deflate, br 8 Referer: https://0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net/login 9 Upgrade-Insecure-Requests: 1 10 Sec-Fetch-Dest: document 11 Sec-Fetch-Mode: navigate 12 Sec-Fetch-Site: same-origin 13 Sec-Fetch-User: ?1 14 Te: trailers 15</pre>	6	Accept-La	anguage: en	-US, en; q=0	9.5					
<pre>8 Referer: https://0a9900bf038cbe8080eb6c2a003b00ce.web-security-academy.net/login 9 Upgrade-Insecure-Requests: 1 10 Sec-Fetch-Dest: document 11 Sec-Fetch-Mode: navigate 12 Sec-Fetch-Site: same-origin 13 Sec-Fetch-User: ?1 14 Te: trailers 15</pre>	7	Accept-Er	ncoding: gz	ip, deflat	te, br					
<pre>9 Upgrade-Insecure-Requests: 1 10 Sec-Fetch-Dest: document 11 Sec-Fetch-Mode: navigate 12 Sec-Fetch-Site: same-origin 13 Sec-Fetch-User: ?1 14 Te: trailers 15</pre>	8	Referer:	https://0a	9900bf038a	be8080eb6c2a	a003b00ce.w	eb-security-acad	emy.net/login		
<pre>10 Sec-Fetch-Dest: document 11 Sec-Fetch-Mode: navigate 12 Sec-Fetch-Site: same-origin 13 Sec-Fetch-User: ?1 14 Te: trailers 15</pre>	9	Upgrade-1	Insecure-Re	quests: 1						
<pre>11 Sec-Fetch-Mode: navigate 12 Sec-Fetch-Site: same-origin 13 Sec-Fetch-User: ?1 14 Te: trailers 15</pre>	10	Sec - Fet cl	n-Dest: doc	ument						
<pre>12 Sec-Fetch-Site: same-origin 13 Sec-Fetch-User: ?1 14 Te: trailers 15</pre>	11	Sec-Fetch	n-Mode: nav	igate						
13 Sec-Fetch-User: ?l 14 Te: trailers 15	12	Sec-Fetch	n-Site: sam	e-origin						
14 Te: trailers	13	Sec - Fet cl	n-User: ?l							
15	14	Te: trail	lers							
	15									

2- Cliquer sur le bouton **Drop** pour annuler la requête. Un message d'erreur s'affiche (ERROR : request was dropped by user) :

3- Dans l'URL de la transaction, remplacez **/login2** par **/my-account**, puis appuyez sur ENTRÉE pour exécuter la requête avec l'URL modifiée. Ensuite, dans BurpSuite, désactivez l'interception et revenez à la page d'authentification. Le défi est alors réalisé.



Défi n°3 : Réinitialisation malveillante d'un mot de passe

Le lien permettant d'accéder au labo est le suivant :

https://portswigger.net/web-security/learning-paths/authentication-vulnerabilities/vulnerabilities-in-other-authentication-mechanisms/authentication/other-mechanisms/lab-password-reset-broken-logic

La fonctionnalité de réinitialisation du mot de passe de ce laboratoire est vulnérable. Pour résoudre le défi, il va falloir réinitialiser le mot de passe de l'utilisateur Carlos, puis se connecter et accéder à sa page via le lien intitulé « mon compte ».

Vos identifiants en tant qu'attaquant : wiener/peter Login de la victime : carlos



LAB Not solved

Home | My account



TO

LIKE

Travaux préparatoires :

1- Accédez à la page d'accueil du défi, puis cliquez sur le lien **My account** en haut à droite. Une fois que la page d'authentification s'affiche, cliquez sur le lien **Forgot password** et saisissez votre identifiant d'attaquant dans le formulaire qui apparaît, sans valider.

	Login
	Usemame
	Password
	Forgot password?
Please enter ye	our username or email
wiener	
Submit	

2- Activez le proxy de BurpSuite (Intercept : on'), puis confirmez en cliquant sur Submit.

Réalisation de l'attaque :

1- Suivez la procédure de réinitialisation du mot de passe en le définissant à la valeur souhaitée. Confirmez chaque étape en cliquant sur le bouton **Forward** de Burp.

Web Security	Password reset broken logic					
Academy 🔬	Back to lab home	Email client				

Please check your email for a reset password link.

Displaying all emails @exploit-0a3600dd034af17f80ca0ce301eb0072.exploit-server.net and all subdomains

Your email address is wiener@exploit-0a3600dd034af17f80ca0ce301eb0072.exploit-server.net

	10	From	Subject	Body
				Hello!
	wionor@ovploit	10		Please follow the link below to reset your pas sword.
2025-04-22 07:03:08 +0000	0a3600dd034af17f80ca 0ce301eb0072.exploit- server.net	reply@0a630054032cf118 80ae0ddd007f0021.web- security-academy.net	Account recovery	https://0a630054032cf11880ae0ddd007f0021.web-s View ecurity-academy.net/forgot-password?temp-forgo raw t-password-token=2gw59pytrqm7t9hhnuvsv8on6hbjb ux0
				Thanks, Support team
	N	ew password		
	N	ew password		
		ew password	ord	
		ew password ••••• onfirm new passw	ord	
		ew password ••••• onfirm new passw •••••	ord	
		ew password ••••• onfirm new passw •••••	ord	

2- Une fois la procédure de réinitialisation du mot de passe terminée, BurpSuite a enregistré toutes les requêtes envoyées au serveur, ce qui permet de les analyser. Dans Burp, accédez à Proxy > HTTP history pour examiner ces requêtes. Recherchez une requête POST associée au changement de mot de passe, incluant le nouveau mot de passe que vous avez saisi.

Inte	rcept HTTP history	WebSock	ets history		Proxy settings				
∇	Filter settings: Hiding CSS, imag	ge and gene	eral binary co	ntent;	matching expression	on POST			
#	Host	Me \vee	URL			Params	Edited	1	Status
684	https://0a630054032cf1188	POST	/forgot-pas	sword	?temp-forgot-pa	~		3	302
688	https://play.google.com	POST	/log?hasfas	t=true	&authuser=0&fo	~			
689	https://play.google.com	POST	/log?hasfas	t=true	&authuser=0&fo	~			
102	https://ogads-pa.clients6.g	OPTIO	/\$rpc/goog	le.inte	rnal.onegoogle.a			-	200
124	https://play.google.com	OPTIO	/log?format	=jsoni	&hasfast=true	~			200
Req	uest					ø	8	۱n	=
<pre>2 Host: 0a630054032cf11880ae0ddd007f0021.web-security-academy.net 3 Cookie: session=zP8Co5nfr7sgxIKDc8dxs5np3Mn4lIFD 4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0 5 Accept: text/html,application/xhtml+xml,application/xml;c=0.9,image/avif,image/webp,*/*;q =0.8 6 Accept-Language: en-US, en;c=0.5 7 Accept-Encoding: gzip, deflate, br 8 Content-Type: application/x-www-form-urlencoded 9 Content-Length: 117 10 Origin: https://0a630054032cf11880ae0ddd007f0021.web-security-academy.net 1 Referer: https://0a630054032cf11880ae0ddd007f0021.web-security-academy.net/forgot-password 7temp-forgot-password-tokem=2gw59pytrqm7t9hhnuvsv8on6hbjbux0 12 Upgrade=Insecure=Requests: 1 13 Sec-Fetch-Dest: document 14 Sec-Fetch-User: 71 15 Te: trailers</pre>									
18 19	:emp-forgot-password-toke new-password-l=certa&new-	en=2gw59j -password	pytrqm7t9h d-2=certa	hnuv	sv8on6hbjbux0&u	sername=	wiener	6	

3- Sélectionnez-

cette requête puis l'envoyer au répéteur de Burp Suite.

4- Sur la page du répéteur, laissez la variable **username** vide, puis comparez le résultat avec celui obtenu lorsque la variable contient une valeur. On remarque que le comportement de la page de retour reste identique. Cela indique que le serveur ne vérifie pas le contenu de la variable transmise dans la requête POST.

Send 🔞 Cancel < 🔻	> * Follow redirection			
Request Pretty Raw Hex		& ≣ \n ≡	Response	Raw Hex Render
<pre>1 POST /forgot-password?temp-fo HTTP/2 2 Host: 0a630054032cf11880ae0dd 3 Cookie: session=zP8Co5nfr7sgx 4 User-Agent: Mozilla/5.0 (X11; 5 Accept: text/html,application/xhtml+x .8 6 Accept-Language: en-US,en;q=0 7 Accept-Encoding: gzip, deflat 8 Content-Type: application/x-w 9 Content-Length: 111 10 Origin: https://0a630054032cf1 11 Referer: https://0a630054032cf11880ae0 emp-forgot-password-token=2gw 12 Upgrade-Insecure-Requests: 1 13 Sec-Fetch-Dest: document 14 Sec-Fetch-Dest: document 14 Sec-Fetch-Dise: same-origin 16 Sec-Fetch-User: ?1 17 Te: trailers 18 19 temp-forgot-password-token=2g new-password-1=certa&new-pass</pre>	rgot-password-toker=2gw59pytrqm7 d007f0021.web-security-academy.n IKDc8dxsSnp3Wn4lIFD Linux x86_64; rv:109.0) Gecko/2 ml,application/xml;q=0.9,image/a .5 a, br ww-form-urlencoded 11880ae0ddd007f0021.web-security ddd007f0021.web-security-academy 59pytrqm7t9hhnuvsv8on6hbjbux0 word-2=certa	t9hhnuvsv8on6hbjbux0 met 20100101 Firefox/115.0 avif,image/webp,*/*;q=0 academy.net .net/forgot-password?t	1 HTTP/2 3 2 Location 3 X-Frame 4 Content 5 6	302 Found -Options: SAMEORIGIN -Length: 0

5- Répétez les étapes permettant à l'attaquant de réinitialiser son mot de passe en capturant les requêtes avec Burp Suite. Ensuite, envoyez la requête POST de réinitialisation du mot de passe vers le répéteur et remplacez le contenu de la variable **username** par celui de la victime. Modifiiez la requpete pour réinitialiser le mot de passe de la victime, puis connectez-vous à son compte. Le défi est alors réalisé.

WebSecurity Password reset broken logic Academy Back to lab description	LAB Solved
Congratulations, you solved the lab!	Share your skills! 😏 🛅 Continue learning »
	Home My account Log out
My Account	
Your username is: carlos	
Your email is: carlos@carlos-montoya.net	
Email	

http://www.reseaucerta.org coso novembre 2025 - v1.0

Update email