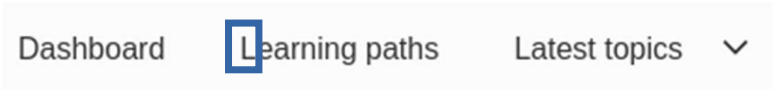
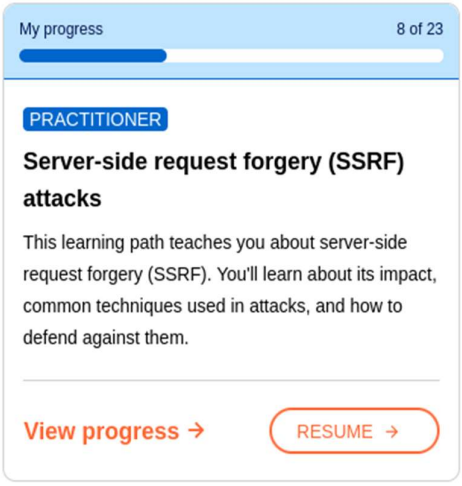
## Défi n°2 : Autre exploitation d'une vulnérabilité SSRF

Une fois connecté avec un compte sur portswigger, depuis la page d'accueil, cliquer sur Academy.


Puis, cliquer sur Learning Path.

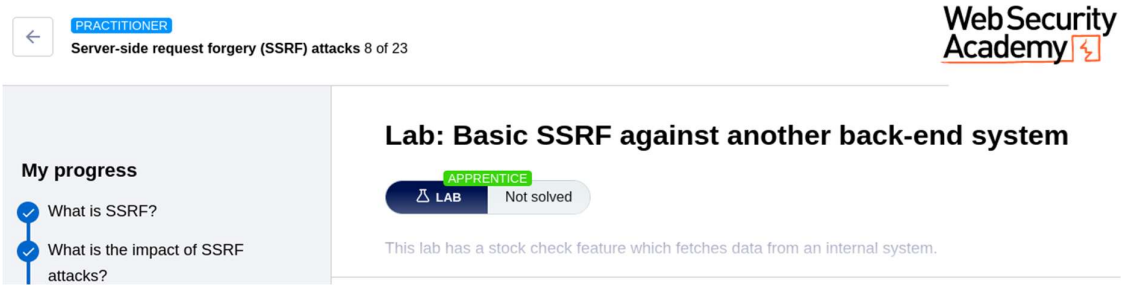



Ensuite, sélectionner le parcours Server Side Request Forgery.




D'autres parcours existent afin de se former en cybersécurité tout en suivant sa progression. Cliquer sur RESUME.

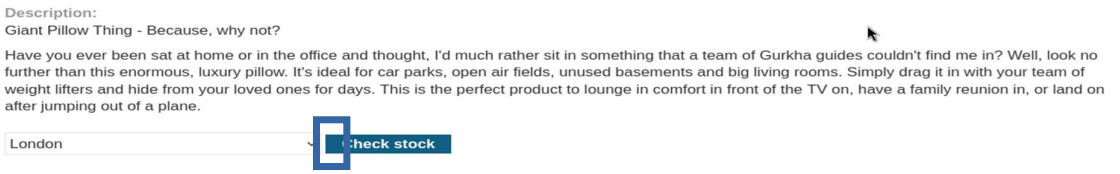
Enfin, avancer dans le parcours jusqu'à l'affichage du labo suivant :



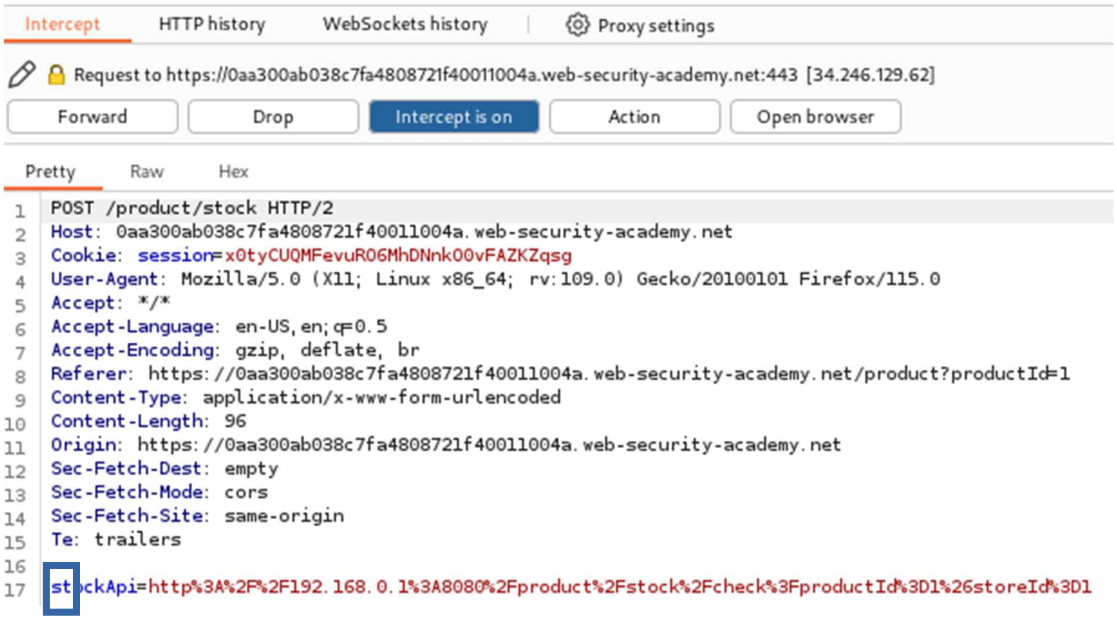

Le lien direct vers le labo est le suivant :

portswigger.net/web-security/ssrf/lab-basic-ssrf-against-backend-system

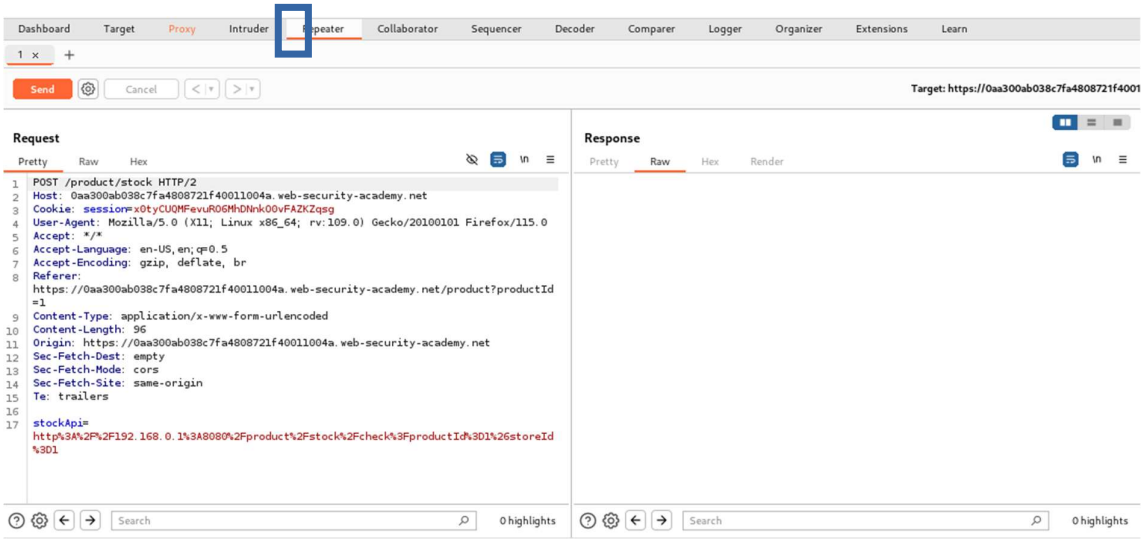
Il faut ensuite cliquer sur le bouton Access The lab. Puis, sélectionner un produit et afficher la quantité en stock tout en ayant le proxy Burp activé.



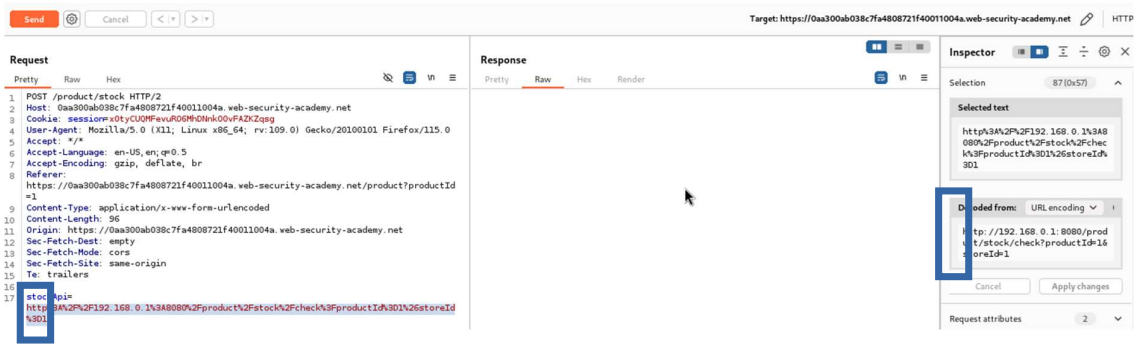

Le paramètre stockApi s'affiche sur Burp comme lors du défi n°1 et contient une adresse IP.



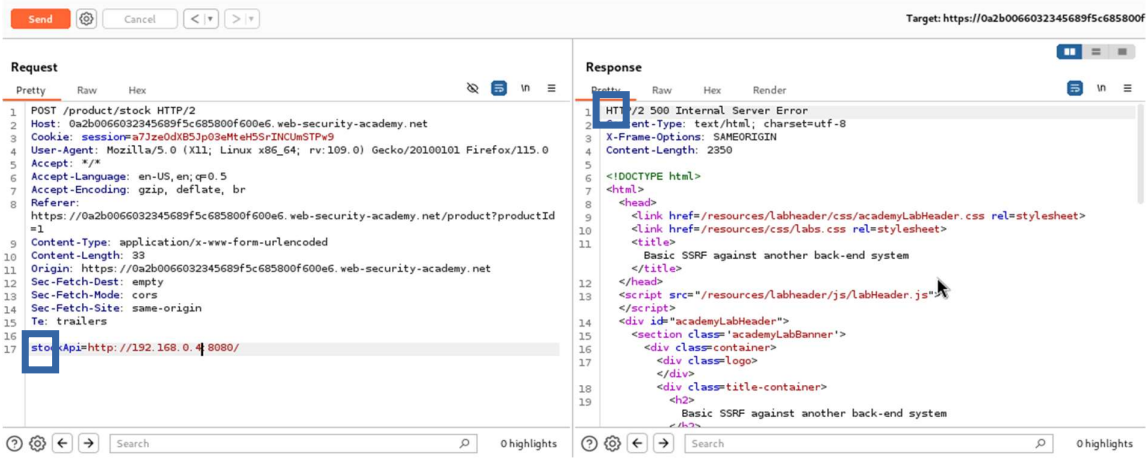

Envoyer vers le répéteur cette page de code contenant le paramètre stockApi.




La sélection de l'URL avec la souris suffit à afficher l'encadré permettant de voir l'URL décodé.



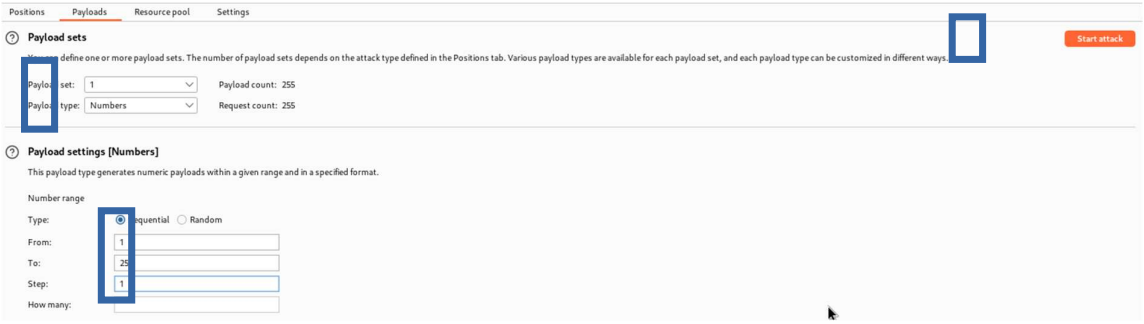
En modifiant manuellement l'URL, on peut essayer une découverte manuelle d'autres instances de serveur.



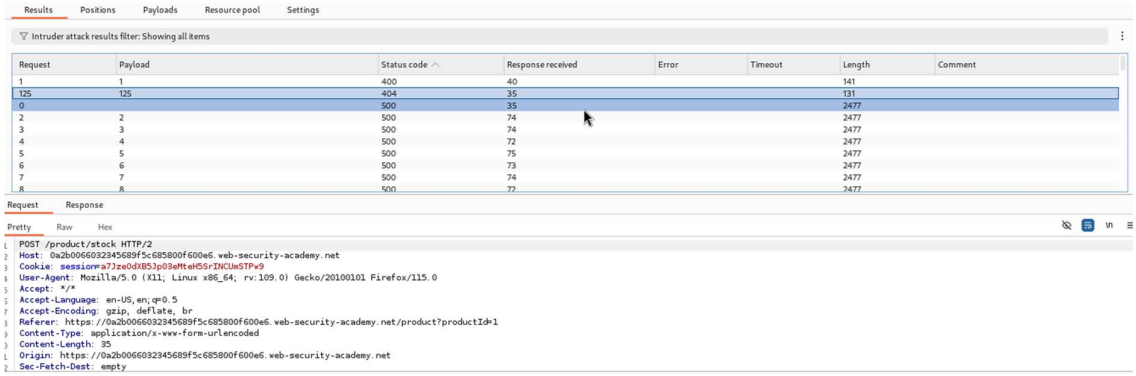
Selon le code de retour du serveur on peut savoir si l'instance de serveur est présente ou non. Pour éviter de faire plusieurs tentatives manuelles, on va automatiser cette tâche en activant le mode sniper.

Afin d'activer le mode sniper, il faut envoyer la requête vers le mode Intruder. Ensuite, il faut sélectionner le 4ème octet de l'adresse IP puis cliquer sur Add.

La configuration du Payload nécessite de choisir une itération de 1 à 255 sur une liste de nombres. Pour cela, il faut aller dans l'onglet Payloads.

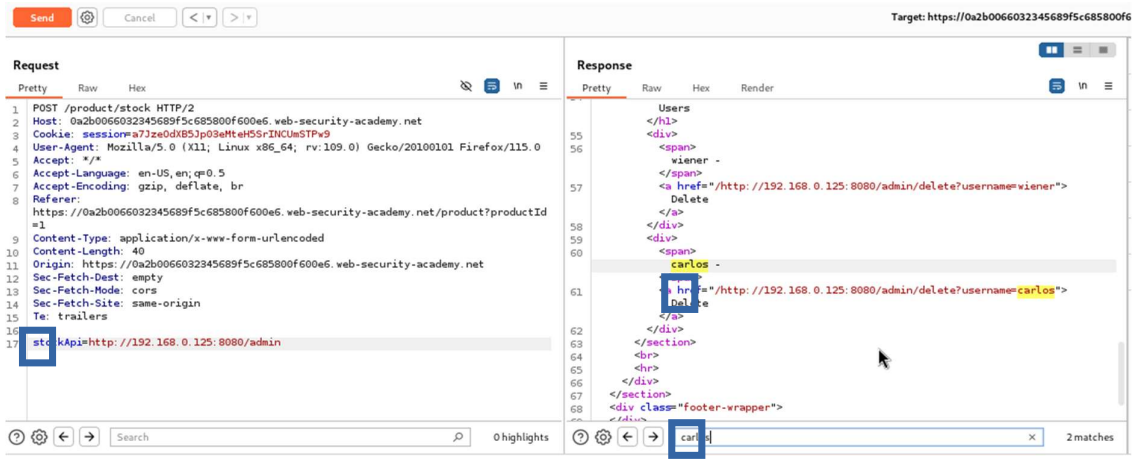



Le résultat du scan permet de voir qu'une page rend le code d'erreur 404 (et non pas 500 pour une erreur côté serveur) sur l'adresse IP se terminant par 125 (192.168.0.125). Pour cela, il faut trier les résultats sur la colonne status.




Il suffit alors d'envoyer la requête correspondante vers le répéteur puis de reprendre la procédure utilisée lors du défi n°1 en testant l'URL suivante :

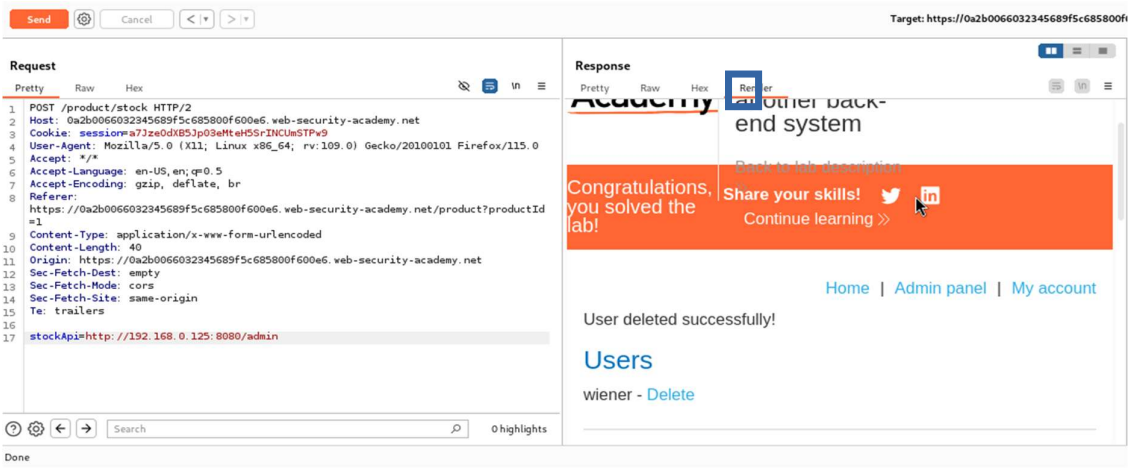
http://192.168.0.125:8080/admin

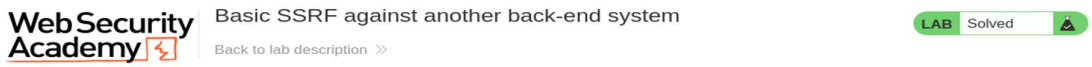



La suppression de Carlos se fait avec l'URL suivant :

http://192.168.0.125:8080/admin/delete?username=carlos

Puis en suivant le redirection, l'utilisateur Carlos est supprimé.


Send
Cancel
Target: https://0a2b0066032345689f5c685800f
Request
Pretty
Raw
Hex
POST /product/stock HTTP/2
Host: 0a2b0066032345689f5c685800f600e6.web-security-academy.net
Cookie: session=a7Jze0dXB5Jp03eMteH5SrINCUmSTPw9
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer:
https://0a2b0066032345689f5c685800f600e6.web-security-academy.net/product?productId=1
Content-Type: application/x-www-form-urlencoded
Content-Length: 40
Origin: https://0a2b0066032345689f5c685800f600e6.web-security-academy.net
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers
stockApi=http://192.168.0.125:8080/admin
Search
0 highlights
Response
Pretty
Raw
Hex
Render
end system
Congratulations, you solved the lab!
Share your skills!
Continue learning
Home | Admin panel | My account
User deleted successfully!
Users
wiener - Delete
Done


Web Security Academy
Basic SSRF against another back-end system
Back to lab description
LAB
Solved