Lab name: Basic SSRF against another back-end system

Severity: High

Lab description:

* This lab demonstrates a **Server-Side Request Forgery (SSRF)** vulnerability in a product details page.
* The application fetches product images from a URL provided by the user.
* The URL is not properly validated, so an attacker can make the server send requests to **internal services** (e.g., http://localhost or http://169.254.169.254).

Impact:

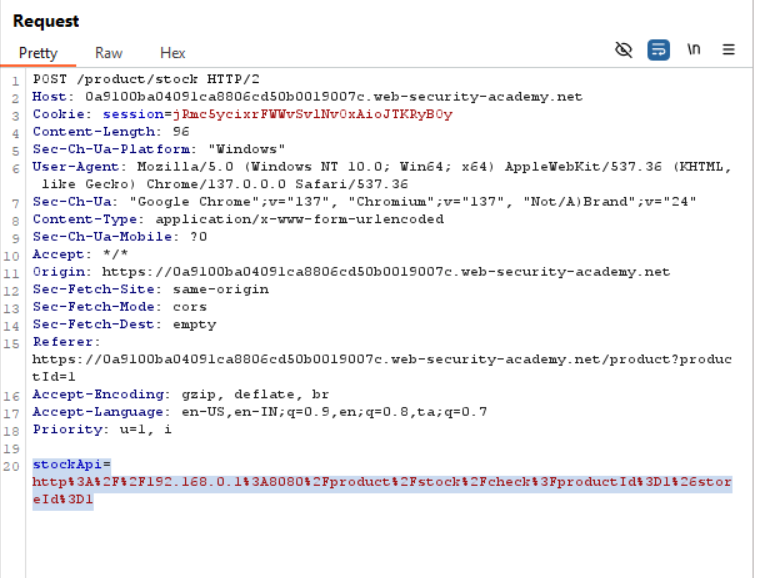
* An attacker can access internal systems not meant to be public, like admin panels or metadata services.
* It may expose sensitive information such as cloud credentials or internal APIs.
* Could lead to **privilege escalation**, information disclosure, or even full system compromise.
* The request is made by the server itself, so it bypasses IP-based firewalls or access controls.
* Attackers can scan internal ports and map the internal network.
* If combined with other flaws, it may lead to **Remote Code Execution (RCE)**.

Recommendations:

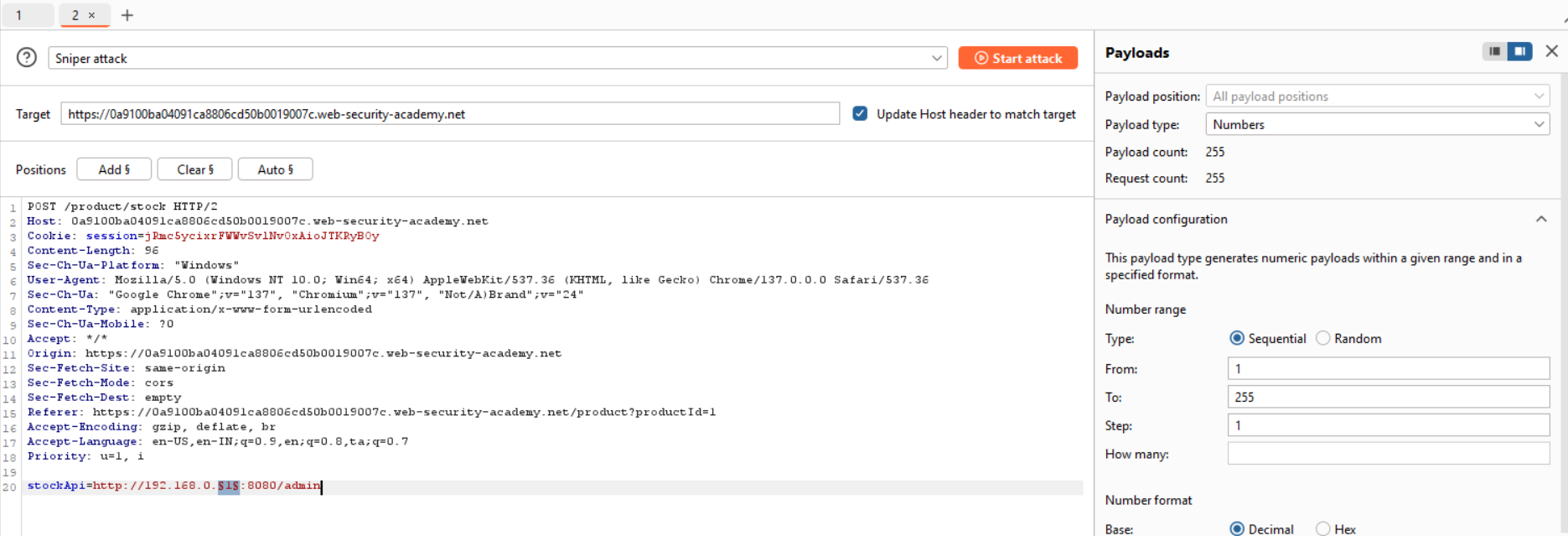
* Validate and whitelist only trusted image URLs (e.g., from your own CDN or domain).
* Block requests to **localhost**, **internal IPs**, and **cloud metadata addresses** (169.254.169.254).
* Use network-level firewalls to prevent the server from reaching internal services.
* Avoid accepting full URLs from users when possible—use ID-based or filename-based systems instead.
* Log and monitor outbound requests from the server for any suspicious activity.
* Apply the **principle of least privilege** to backend services so that even if SSRF occurs, the impact is limited.

Steps to reproduce:

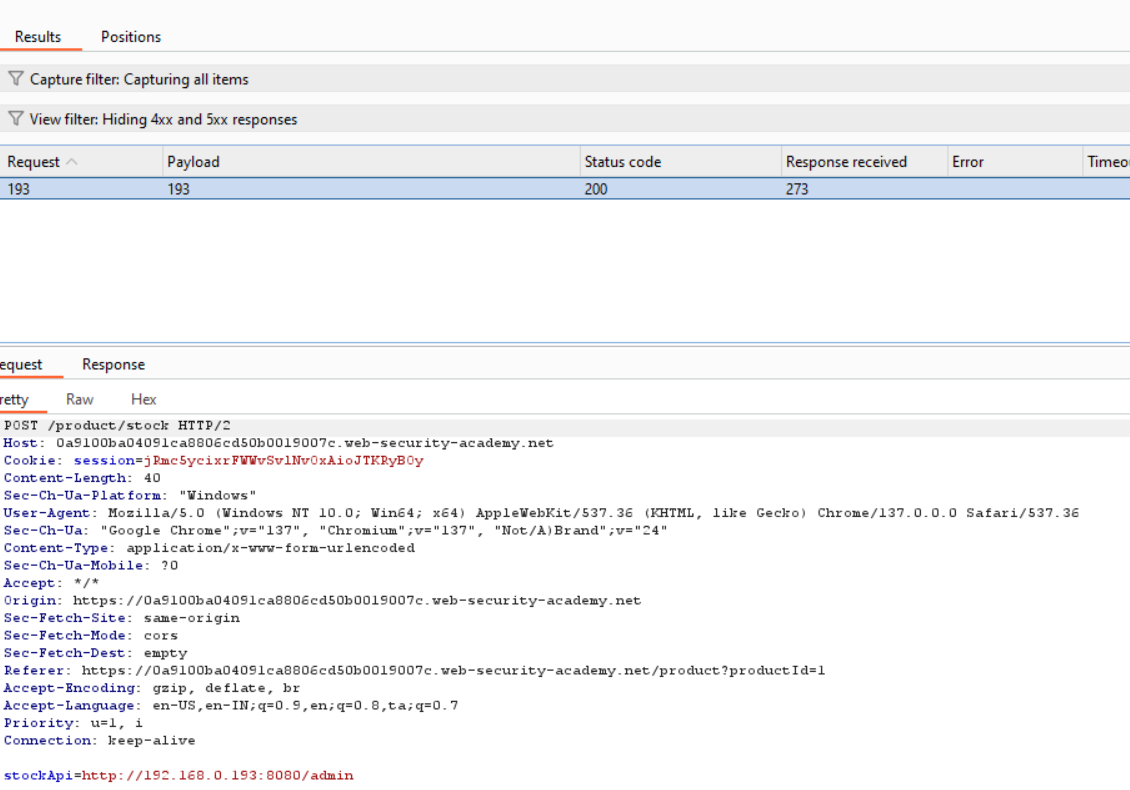
* 1. Access the lab and visit a product from the store 🡪 check store button.
  2. Intercept this request and send the POST request to intruder.



* 1. Change the stockApi parameter to <http://192.160.0.1:8080/admin>, and add ‘1’ from the octet to the payloads. Select type as numbers and range from 1 to 255. This will send 256 requests by trying all the IP addresses. Start the attack.



* 1. The payload with status code 200 refers to the successful payload. In this case, it is 192.160.0.193.



* 1. Repeat the request in browser, then the admin page is accessed. Now if carlos user is deleted, the lab is solved.

