CSOa challenge lab

Your goal for this challenge lab is to acquire remote root access at the target machine. Through your journey, you will find and exploit a chain of vulnerabilities, every one of which granting you deeper access to the machine. Please document steps you've taken to find and exploit these vulnerabilities as it will serve as proof you solved the lab. Send the documentation (.pdf, .txt or .md, 2-4 pages) to iletavay@fit.vut.cz with CC to veselyv@fit.vut.cz with subject `[CSOa] Challenge Lab submission`.

Local environment setup:

1. Download the target machine as a Virtual Machine (VM) and import it into the VirtuaBox (tested with version 7.0.14).
2. VM is configured to use NAT networking with TCP ports 22/2222, 8000/8000 and 8443/8443 forwarded. If any of 2222, 8000, 8443 ports collide with any other services running on your host, feel free to choose different (host) ports and use them instead. Furthermore, you can switch VM's network adapter to bridged mode, which will result in the VM acting as a separate network host on your LAN with its own IP address.
3. Run the VM and interact with it only via a network, as you would with a real remote server. Don't waste your time with attempts to logging into the local VM console.

Cloud environment

If you cannot host the VM on your machine (for example, because it is an ARM machine), you can request a VM instance at a faculty's virtualization server from iletavay@fit.vut.cz. You will be provided with a Wireguard VPN configuration profile using which you will be able to access your VM instance at IP address `10.99.0.1`.

Verification

Once you start the local VM or get access to a cloud one, you can try to access a website running on the VM via `http://localhost:8000` (for the local VM) or `http://10.99.0.1:8000` (for the cloud VM). You should see the *Welcome to nginx!* page.

Objectives

1. Enumeration: gather information about the web application (what is its overall functionality, what technologies are used and how the web application could be implemented, what user inputs are processed by the web application, etc).
2. Service exploitation: find and exploit service's vulnerabilities resulting in command execution.
3. Privilege escalation: acquire access to the **host machine's root user** (identifiable with `root@ubuntu-jammy` prompt) from the unprivileged user.

Resources

1. https://www.smashingmagazine.com/2023/06/popular-devtools-tips/#2-edit-and-resend-network-requests

2. https://github.com/swisskyrepo/PayloadsAllTheThings
3. https://gtfobins.github.io/

Spoiler alert - Additional keywords to search the study materials for (in no particular order): docker, SQL injection, file upload vulnerabilities, local file inclusion, reverse shell, password cracking, container escape, HTTP/S, linux privilege escalation, virtual web hosting