Honeypot project guidelines

Overview

This document contains all information about the project for the course Websecurity & Honeypot. The project is made in groups of maximum 3 people. The groups are assigned using Leho. Read this document carefully for the assignment requirements, deliverables and rules. A honeypot is a computer security mechanism set to detect, deflect, or, in some manner, counteract attempts of unauthorized use of information systems.

Generally, a honeypot consists of data (for example, in a network site) that appears to be a legitimate part of the site, but is actually isolated and monitored, and that seems to contain information or a resource of value to attackers, who are then blocked.

Project Assignment

Setting up & securing Honeypot server

Secure a web application and webserver without limiting the functionality for users. Each group gets a virtual machine to install a webserver and web application. This web application is accessible from inside the Howest network for anyone with a browser.

Each web environment is required to have the following features:

- Users can *secure* login
    - Without email verification, captcha or 2FA
- Users can register a new account
- Upload an avatar
- 5 'Honeypot' challenges crf exam Web Pentesting
    - SQLi, XSS, Broken Access Control, ...
- *Secure* admin panel with user management
    - Overview of users currently logged in
    - Overview of all users
    - Enable/Disable users
      Note: The web application can be an existing product or own made.
- Extensive secured logging environment (Kibana)
    - Extended logging
    - Dashboards (overview and per challenge) with *at least* the following items:
        - Attempts per user
        - Payloads
    - Authentication

    - create a user `lecturer` with password `ThisIsTheM0stSecureKibanaEver`

All code/configuration scripts are maintained in the Gitlab repositories. The repository has an extended README.md file with information about the project.

Securing your honeypot

As an administrator of the honeypot the group is responsible for installing, configuring and securing the web server and the web application(s). The web server and web application needs to be running the rest of the semester and needs to be maintained.

The group is responsible for the continuity of the system and trying to deflect all hacking attempts or at least analyze the attempts and document them in your journal (until the end of the semester – see deadlines!). If your honeypot gets compromised, an analysis and assessment of the damage needs to be done and documented in your log. As an administrator you can also pentest your own system in order to harden the system.

- No limitations can be applied on the IP address
- The usage of a packet filter IS NOT allowed
- It is forbidden to install infrastructure services like DNS, DHCP, terminal server... or any other software that might disrupt the network or other services of Howest.

Note: You can use authentication or IP address restrictions during the first stage of the project.
Once arrived at the first deadline (see further), these restrictions have to be removed! The account assigned to each group will be given after the server is online. This account cannot be used by someone outside the group. Logging into your server is done using remote desktop or SSH. You log out or disconnect, **DO NOT SHUTDOWN THE MACHINE!**

Pentest Honeypot servers

Each group also has to pentest the honeypots from the other groups. The groups should not exchange information with each other. You should be able to solve the challenges of the other groups.

During the pentest the following requirements must be applied.

- Test the web application and web server for configuration mistakes and bugs/exploits in the software
    - Not limited by the challenges but these are the main goal
- Use the techniques learned in Web Pentesting
    - Not limited by that!
- It's **not allowed** to attack the **Kibana** or logging environment
- It’s under no circumstances allowed to use the honeypot servers to attack other systems or to disrupt the network of Howest.
- It’s forbidden to attack Howest authentication or other infrastructure services. The lecturer or ICT Helpdesk can shut down a Honeypot without prior notice. Abuse can result in stopping the honeypot project.
- The remote desktop interface (SSH) cannot be used in your attacks. The root account is for the lecturer and ICT Helpdesk. They cannot be targeted.
- Social engineering the helpdesk or the lecturer is out of the question.

Important for extra points:
Somewhere during the project, the lecturers will hide an attack in the logging files, show during evaluation what attack took place and at what time. *Note: it's obvious in the log when the attack is by the lecturers*
When found a bug in the environment of another group create a write up of the attack and send this for evaluation to the lecturers. The write up must contain the following items:

- Title
- Sevirity
- Type of weakness
- Description
- Summary
- Steps to reproduce

The lecturer will evaluate the proof of concept and contact the student and the hacked group.
Note: only the first accepted proof of concept will get extra points.

Evaluation

The Honeypot project is 50% of the total grade of this module. Evaluation is based on the following items:

- Check of the server with the requirements at deadline 1
- Scripts used to setup the environment at deadline 1 on Gitlab
- Dashboard data at deadline 2
- Live demonstration at deadline 2

Practical

Deadlines

1. Thursday 24th November 10h00: The webserver and application have to up and running and secured by then. From this moment, you are allowed to attack the other honeypots.
    - *Evaluation is based on the webpages reachable, logging and the Ansible script on Gitlab*
    - Friday 2th December: An announcement will be sent to all students of Applied Computer Sciences. From that moment they are allowed to attack the honeypots as well.
2. Wednesday 14 December 13h30: live demonstration of the environment.
    - *Evaluation is based on the demonstration with these topics*
        - Demo the Kibana data
        - What did you find in the data?
        - Are there things you didn't find in the dashboards?
        - What did you find on other servers?
        - What is the best challenge on the other servers and how did you attack that?
        - What would you change on the project?