Brandenburg University of Technology
Cottbus-Senftenberg
IT Security Group

Exercise Class
“Introduction into Cyber Security”
Winter Term 2018/2019

Introduction into Cyber Security – Web-Security using OWASP WebGoat

Deadline: 30th January, 2019

Topics

This experiment deals with common vulnerabilities encountered in the World Wide Web. You will educate yourself in the basic vulnerabilities, like the abuse of missing input filtering and/or user identification, as well as the more advanced topics like SQL-Injection and Cross-Site-Scripting (XSS). If you encounter difficulties, study the given material carefully and do not give up too quickly – tenacity is an important skill for security experts.

1 Preparation

To provide the World Wide Web, web servers have to supply HTML pages to users. For this purpose, the servers often rely on the LAMP (**L**inux-**A**pache-**M**ySQL-**P**HP) stack. Since the architecture of the web servers is often so similar, this makes it quite easy for device attacks that apply to a considerable amount of machines on the Internet. Among many others, the reliance on HTML and LAMP has created a particular variety of very common vulnerabilities.

To gain an understanding of the latter you will be using the web application ‘WebGoat’. It was developed especially for the purpose of security education and implements the most common vulnerabilities of web applications.

In this laboratory, the machine `luke` is running a web server on port 8080 which hosts ‘WebGoat’. This application can be accessed using a web browser and will teach you a selection of common web hacking vulnerabilities. Since “WebGoat” intentionally opens up vulnerabilities on the host machine, `luke` accepts connections from the server `mouse` only. Therefore you have to connect to `mouse` first, and then from there to `luke`.

1.1 Connection to luke

The experiment can be carried out either in the computer lab (room 0.02, choose Linux as OS) or on your own personal computer. All that is needed are a web browser (recommend to use firefox) and a SSH client.

Since “WebGoat” intentionally opens up vulnerabilities on the host machine, `luke` accepts connections from the server `mouse` only. Therefore you have to connect to `mouse` first, and then from there to `luke`.

Keep in mind, that the server names can only be resolved in the institute network – from outside you may reach `mouse` under the domain `mouse.informatik.tu-cottbus.de`.

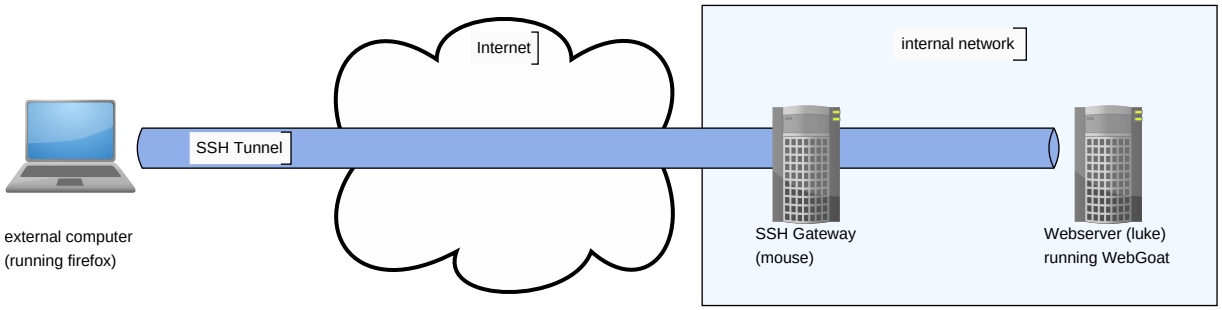

Figure 1: Laboratory network – connection is only possible by tunneling through `mouse`

To create a connection that you can use in your web browser, it is recommended to create a SSH tunnel through `mouse`. For this you can either use

- the `ssh` command line application (recommend on Linux)
- the application ‘PuTTY’, [4] (recommend on Windows)

Tunneling can be activated for `$ssh` by using the `-L` command line argument. For more detailed information regarding the topic ‘remote port forwarding’, various resources, like [7], [2], are available online.

1.2 Working with WebGoat

After you have got access to the web-server `luke`, you can log into the web-interface of "WebGoat" (`.../WebGoat/attack`), using the provided credentials. You will see the landing-page of WebGoat. You can find the tasks to solve in the side-panel on the left side of the page. The tasks can be solved by normal interaction with your web browser, using the provided buttons and edit fields of the website itself and/or using the URL line.

Since WebGoat was developed for educational purposes it offers some practical options which you may want to use during solving this task. You can find these options in the top-panel of the "WebGoat" interface.

- `Hints` – If you encounter problems during solving a specific task you can get additional hints here
- `Show Params` – Using this option you can take a look at the data and parameters containing the send HTTP-request.
- `Show Java` – By this option you can take a look into the source of the server-side Java applications.

If a task is solved successfully it will be marked green in the side-panel. The current state of the exercise can be monitored in the `Report Card` under the `Admin Functions` tab. This exercise is successfully passed if and only if all tasks listed in the category "Normal user lessons" on the `Report Card` are passed, i.e. marked green.

Additional helpful information about the SQL-Injection can be found in the references [5], [6]. Further, to solve the task "Spoof an Authentication Cookie" of the "Session Management Flaws" category, an additional tool to edit cookies may be useful. We recommend to use the browser plug-in "Cookie Manager" for firefox [3].

2 Main Task

Setup the connection to the machine `luke` to get access to "WebGoat". For details refer to the guide provided in section 1.1. Open WebGoat (see section 1.2) and work through all of the tasks listed under "Normal user lessons" int the `Report Card`. The needed credentials to log into the WebGoat-interface are provided to you by the moodle platform. Once you are done, send an notification email to ziemator@b-tu.de (Please use the subject "Solution:ics_practical_task_03").

The notification has to be done before the end of the deadline.[1]

Good luck!

References

[1] *A comprehensive interactive guide to many WWW vulnerabilities*. 2019. URL: https://www.hacksplaining.com/.

[2] T. Blog. *SSH Tunnel - local and remote port forwarding explained with examples*. 2019. URL: http://blog.trackets.com/2014/05/17/ssh-tunnel-local-and-remote-port-forwarding-explained-with-examples.html.

[3] *Cookie Manager*. 2019. URL: https://addons.mozilla.org/de/firefox/addon/a-cookie-manager/.

[4] *PuTTY*. 2019. URL: https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html.

[5] *SQL - Structured Querry Language*. 2019. URL: https://en.wikipedia.org/wiki/SQL.

[6] *SQL Reference Manual*. 2019. URL: https://dev.mysql.com/doc/refman/8.0/en/.

[7] *SSH Secure Shell*. 2019. URL: https://linux.die.net/man/1/ssh.

[8] Symantec. *Five Common Web Application Vulnerabilities*. 2019. URL: https://www.symantec.com/connect/articles/five-common-web-application-vulnerabilities.

[1] Note: By sending the notification e-mail we will save your current state of the exercise by backing up your `Report Card`. These information will be used for the grading of this task.