Ciências ULisboa | Informática

Metodologias Ofensivas e Defensivas de Cibersegurança (MODC)

Mestrado em Segurança Informática

Mestrado em Engenharia Informática

Mestrado em Informática

2023/2024

Project

1. Objectives

The main objective of this project is to give the students the experience of the work a security team must face in a company, for the different valences that the team's members should have. The project is split into threefold objectives. First, it allows students to discover real vulnerabilities in application scenarios and generate and disseminate threat intelligence about their findings. Second, it permits students to set up a threat intelligence platform to help them in incident response, vulnerability management and risk analysis activities. Third, it lets students try to find adequate machine learning procedures to obtain precise answers in the cybersecurity field.

2. Assets to Monitor and Security Team

The hypothetical scenario of the company to be monitored consists of three applications (assets): two internal and one external (i.e. third-party software). The internals are the application that will build by the group (`i1`, see Section 3.1.1) and an application provided by the professor (`i2`, see Section 3.2). On the other hand, the external is an application produced by some other group (`e1`). The security team only knows the functionality of the application created by their own group (company). For the other two applications, their source code is available, but the group needs to analyze it if it would like to know how applications work internally (something that may be necessary during the project).

The security team is composed of three members that will play the roles of PT (pentester), SOC & IR (incident response), VM & RA (vulnerability management and risk analysis), and DS (data scientist). All members have skills in these fields; however, a member is responsible for a field. Also, the team has programmer skills for developing (web) applications and scripts.

Note that a member is responsible for a field, does not mean he will make all activities regarding this area! On the contrary, he will define the activities to be performed by the team, including itself, and ensure their execution and deliverables in due time.

3. Technical Details

The project is divided into three parts that are described next. Each part can comprise diverse phases, where their execution can not be sequential, i.e., phases of different parts can be executed simultaneously.

3.1 Part I – Application development (4 points)

3.1.1 Phase 1 - Application development

In this part, the groups need to develop a *photo-sharing* or a *message-sharing* application, which supposedly needs to be made secure. The assumption is that this application will be later deployed in an environment where adversaries might perform attacks of various kinds, for instance, by trying malicious actions on the network or experimenting with malicious clients. Also, security teams will check the security of the applications the company have deployed.

The application scenario should be made realistic. Groups have complete freedom to select the programming language to develop the application and integrate existing file or message-sharing modules. For instance, some groups may choose to develop their version of the application, taking advantage of building blocks already available on the web as open source. The service may be implemented by one server or several servers. For a more realistic scenario of malicious threats that can be present in companies, the application should contain **at least 3 vulnerabilities** adequately identified. The vulnerabilities that were inserted **must have been tested** to make sure that they can be attacked successfully.

The implementation of the application scenario is to be done using virtual machines, where the various components are deployed. Therefore, groups must ensure that their applications run on the virtual machine of the course.

Groups must give their own application a name. **The application's creators must be anonymous**, i.e. the application must not contain any group identification.

Outputs and deliverables:

- Software: A zip file with the application's source code, which will be made available to the other groups.
- Application manual: A manual of the application containing a brief resume of its functioning and installation. The manual has a number of maximum pages: **3 pages**.

- Vulnerabilities definition: A report describing the three vulnerabilities that were inserted in the application. It also needs to describe how these vulnerabilities were tested to ensure they can be exploited (NOTE: you do not need to produce an exploit, but somehow give an idea of how another group could confirm that it exists). The report has a number of maximum pages: **5 pages**.
- Deliverable: 12.Apr, at 23H59 in Moodle
- Mark value: 4 points

3.2 Part II – Vulnerability Discovery and Response (11 points)

As specified before, the security team will **monitor three applications**: two internals (the one built by their own group (`i1`) and another provided by the professor (`i2`)) and one external (`e1`) (built by other group). Also, the security team will perform **pentesting of three applications** (`i2`, `pt1`, and `pt2`).

`i1` internal application:
`i1` is the application built by the group.

`i2` internal application:
`i2` is the "Vulnerable-Web-Application" application from OWASP (https://github.com/OWASP/Vulnerable-Web-Application). The group have to download and install it, following the indications on the webpage. If the group chooses to install the application on the course VM, the way to do it is as follows:

- After downloading the application (e.g., to the `Documents` directory) and unzipping it there, it is better to rename the application folder to one shorter, for example, `vuln_wa`;
- Copy this folder to the apache web server
  ```
  sudo cp -R /home/ss/Documents/vuln_wa /opt/lamp/htdocs
  ```
- Move to the `/opt/lamp/htdocs` and execute the command
  ```
  sudo chmod 775 -R /opt/lamp/htdocs/vuln_wa
  ```
- To make the configurations in the `php.ini`, the file is located at `/opt/lamp/etc/php.ini`
- After all these steps, in a browser, use the URL: `localhost/vuln_wa/`

`e1` external application:
For the external application, the group will receive an application from other group, which will be assigned by the professor.

`pt1` and `pt2` applications:
The group will receive two applications from the other two groups, which will be assigned by the professor. Note that these applications are different of `e1`, and therefore, the group will receive three applications from the other groups (`e1, pt1,` and `pt2`).

3.2.1 Phase 2 – Pentesting (PT)

The group will perform pentesting over the applications `i2`, `pt1` and `pt2` using **at least 2 pentesting tools in each one**. The group is free to choose the tools for pentesting (scanners, fuzzers,…) but must uniformize the results of the tools in a single report.

Outputs and deliverables:

- Report: A report indicating the tools the group used, detailing the important characteristics they have for the target applications. Also, the report must contain the results of the tools, but in a single way, i.e., with uniformization of results of different tools. The report has a number of maximum pages: **5 pages**.
- Mark value: 4 points

3.2.2 Phase 3 – SOC & Incident Response (SOC & IR)

This phase comprises three steps.

Step 1 – Create a cyber threat intelligence feed

The group will create its own feed with the results obtained from **Phase 2** to make it available for the other groups. The feed must be a CSV file, with columns defined as the group considers important and lines with the vulnerabilities the group have discovered. The feed's name should be in the format `feed_modc_xx`, where `xx` is the group identification.

The group must ensure that its feed work correctly in the threat intelligence platform (TIP) chosen, i.e., its feed is correctly added to the TIP.

Outputs and deliverables:

- Report: A report indicating how the CSV file is composed, i.e., its columns. The report has a number of maximum pages: **1 page**.
- Feed: the feed containing the results of pentesting.
- Deliverable: 03.May, at 23H59 in Moodle
- Mark value: 1,5 points

Step 2 – TIP and feeds ingestion

In this step, groups must install a TIP for ingesting the different feeds from all groups. Groups are free to choose the TIP they consider since they ensure their TIP can receive and ingest the feeds. Otherwise, groups can use MISP (https://vm.misp-project.org/latest/).

Afterwards, groups have to add to the TIP all feeds that the professor will provide. For MISP, you can manage feeds using the MISP tutorial (https://www.circl.lu/doc/misp/managing-feeds/#adding-feeds)

Step 3 – Use CTI for incident response

This step aims to use the CTI provided by feeds to help the security team in the monitoring task. Therefore, the goal is:

1) Check if the other groups were able to discover the vulnerabilities inserted in the application built by the group (application `i1`).
2) For application `i2`, check if the other groups discovered other vulnerabilities than the group discovered.
3) For application `e1`, verify which vulnerabilities it presents.

Different feeds can report the same vulnerability of a given application. Hence, it is important to classify these incidents and correlate them so they can appear connected in the TIP.

Also, for the `e1` application, the security team must define the remediation measures to prevent the company from being attacked.

Outputs and deliverables:

- Report: A report responding the points 1), 2) and 3) and presenting the remediation measures the team defined. The report has a number of maximum pages: **5 pages**.
- Mark value: 2,5 points

3.2.3 Phase 4 – Vulnerability Management & Risk Analysis (VM & RA)

In this phase, the group must assess the real impact that the vulnerabilities found in the `e1` application have on the company, analysing the risk if they can be exploited. In addition, groups must indicate how such vulnerabilities should be remediated, and present a patch for them.

Outputs and deliverables:

- Report: A report responding the risk analysis made and the vulnerability remediation process. The report has a number of maximum pages: **5 pages**.
- Mark value: 3 points

Report of Part II:

- Report: A single report containing the specified reports of **Phases 2, 3 (Step 3)**, and **4**.
- Deliverable: 22.May, at 23H59 in Moodle

3.3 Part III – Dataset and Machine Learning procedures (5 points)

3.3.1 Phase 5 – Dataset selection

In this phase, groups need to select a dataset associated with cybersecurity, i.e., where cyberattacks can appear. The selected dataset will be used to employ machine learning methods in order to obtain the best and have better results.

The dataset should not be too small in both dimensions, i.e., in size (number of instances) and in number of columns and should be available under a csv file.

The group can opt by creating its own dataset, collecting data from their network attacks (e.g., when proceed with pentesting), from the systems that register their activities (e.g., web server logs, system logs, application logs), or can opt by using an already existing dataset.
The following links are repositories of existent datasets that groups can use, but are not limited to:
https://www.kaggle.com/datasets
https://www.unb.ca/cic/datasets/index.html
https://elitedatascience.com/datasets
https://github.com/awesomedata/awesome-public-datasets

Outputs and deliverables:

- Dataset characterization and ML goals: A report presenting the characterization of the dataset (topic, HDD size, number of features, number of instances, URL) and what the group intent to do over the dataset and what expect to solve/obtain. The report has a number of maximum pages: **1 page**.

3.3.2 Phase 6 – Machine Learning procedures

The group will define the ML procedures it considers to be experimented with the dataset it chose, analyze their results and make conclusions. The group can also use the data provided from **Part II**, case it considers relevant for the analysis of the dataset.

The major goal of using ML in this part is to detect threats. The group can use a single type of ML (e.g., supervised), but it is not limited to. The group is free to use any type and combination of ML methods, since the goal is to get the best method with precise results.

Outputs and deliverables:

- ML procedures and results: A report presenting the objectives of the work, detailing and justifying the ML procedures used by the group, their results and a dis-

cussion of the obtained results. The report has a number of maximum pages: **5 pages**.

- Mark value: 5 points

Report of Part III:

- Report: A single report containing the specified reports of **Phases 5** and **6**.
- Deliverable: 22.May, at 23H59 in Moodle

4. Evaluation

Students will be evaluated by the following criteria:

- The selected application and the level of realism that is achieved
- The type of vulnerabilities that were inserted and how realistic they are
- PT: the experiments that were performed and how effective they were at finding the vulnerabilities
- SOC & IR: the form CTI was used and integrated in incident response
- VM & RA: the risk analysis made for the real impact and how effective were the vulnerability remediations
- ML: how accurate and precise the ML methods were for the dataset chosen
- **Reports (readability, presentation, detail of the explanation, included discussions, correctness)**
- **Project discussion**

5. Deliverable Dates

Part	Date	Phase
I	12.Apr	1
II	03.May	3, Step 1
II	22.May	2, 3 (Step 3), 4
III	22.May	5, 6