COURSEWORK SPECIFICATION
ECM2426 – Computer and Network Security
Module Leader:
Achim Brucker and Abdelkhalik Mosa
Academic Year: 2024/25

Title: Continous Assessment
Submission deadline: 2024-12-13

This assessment contributes **30%** of the total module mark and assesses the following **intended learning outcomes**:

- Module Specific Skills and Knowledge:
  - Demonstrate understanding of the concepts, issues, and theories of cryptography and security;
  - Demonstrate theoretical and practical knowledge of security technologies, tools, and services;
  - Gain practical experience of developing solutions to networks and computer security challenges.
- Discipline Specific Skills and Knowledge:
  - Show an awareness of the need for network and computer security;
  - Demonstrate good design and development skills.
- Personal and Key Transferable / Employment Skills and Knowledge:
  - Demonstrate practical knowledge of current security methods and tools.

This is an **individual assessment**, and you are reminded of the University's Regulations on Collaboration and Plagiarism. You must avoid plagiarism, collusion and any academic misconduct behaviours. Further details about Academic Honesty and Plagiarism can be found at `https://ele.exeter.ac.uk/course/view.php?id=1957`.

You are requires you work on a virtual machine that is assigned to you individually, and to submit your solution in a very specific format. Read the instructions carefully, it is your responsibility to ensure that your submission adheres to the specified format. Please ensure you read the entire document before you begin the assessment.

Questions about this course work should be asked, in person, before or after the workshops.

This course work consists out of six questions.

Instructions

This continuous assessment covers topics taught in the module ECM2426 “Computer and Network Security”. The continuous assessment focuses on your understanding of the practical aspects of security that you acquired during the labs, lecture, and practical exercises in the lecture notes.

The structure of this continuous assessment follows the structure of the lecture. The submission deadline is towards the end of the term, but you should start immediately to work on this course works: it has been designed as **continuous assessment**. Each exercise denotes a week, in which you should be able to solve it (after completing the workshop on Monday of the same week).

System Access and System Resources

For completing this course work, you need to use a virtual machine (VM) that is provided to you. The VM is hosted on Azure Labs, i.e., the same system that we are also using for the workshops:

- Name of the VM: `2024-ecm2426-ca`
- User: `lh`
- Password: `Initial1`

Do not re-image the VM. MS Azure Labs offers to re-image the VM. Do not use this feature: if you do, **all solutions already obtained will become invalid**, and you will need to re-start the course work from scratch.

Note that access to the cloud system is expensive and billed “by the minute”. Thus, please always switch VMs off after you finished your work. To mitigate the risk of “cost explosion”, the VM

1. is configured to shut-down after 15min of inactivity. Please save you work regularly to avoid loosing work due to an automatic shutdown.
2. Has a maximum number of hours assigned. This is not intended as a limit of hours that you are allowed to work on this coursework. We will monitor the time left for each VM and increase if when the limit is reached. Furthermore, you can request more time by sending am email (the body of your email can be left empty) with the subject “`ECM2426 CA: Please check VM time`” to `a.brucker@exeter.ac.uk`. The deadline for such requests is the last Friday before the submission deadline, precisely:

   2024-11-29T17:59:59 GMT

   After this deadline, request will be processed on a “best effort”-basis, but there is no guarantee that they are processed before the submission deadline.

Referencing and Academic Conduct

This is an individual assessment. Hence, **you are not allowed to discuss solutions of this course work, or instructions how to obtain solutions, with others**. This includes discussions on WhatsApp groups, Discord channels, or posting questions on Stack Overflow. The university requires you to cite the work of others used in your solution and to include a list of references. You must avoid plagiarism, collusion and any academic misconduct behaviours. For further information and guidance, please take the self-learning "Academic Honesty and Plagiarism".

Furthermore, **you need to work on the VM assigned to you**. Not following this rule, will also be considered as academic misconduct.

Marking Scheme

The marks for the individual questions are denoted for each (sub)-question. You might get partial marks awarded. In addition, please note:

- You need to work on the VM assigned to you and only to you. Some tasks result in solutions that are unique to your VM. Hence, **solutions obtained on a different VM will result in zero marks.**
- Your submission needs to follow a specific format, detailed in the next section. **Not following this format will result in zero marks**. As part of the VM, a script is supplied that does provide a basic check of the required format. You are encouraged to make use of this script. If this script cannot recognise your solution/submission, it will likely result in zero marks.
- If you re-image your VM (you should not do this), only the solutions obtained on the latest version of your VM will be valid. Solutions obtained on earlier versions will receive zero marks.
- You need to complete the GenAI declaration that is included in the spreadsheet you need to submit (this is part of the provided template). For more details, read the following two sections very carefully. **Not completing this section will result in a penalty of 10 marks.**

Use of GenAI Tools

The University of Exeter is committed to the ethical and responsible use of Generative AI (GenAI) tools in teaching and learning, in line with our academic integrity policies where the direct copying of AI-generated content is included under plagiarism, misrepresentation and contract cheating under definitions and offences in TQA Manual Chapter 12.3. To support students in their use of GenAI tools as part of their assessments, we have developed a category tool that enables staff to identify where use of Gen AI is integrated, supported or prohibited in each assessment. This assessment falls under the **category of AI-supported**. **You are required to submit a declaration explaining, in detail, the use of AI tools as part of your submission.**

Submission

The submission consists only out of one file in **OpenDocument Spreadsheet** format (`.ods`). This file format is, for example, supported by LibreOffice (`https://www.libreoffice.org/`). LibreOffice is installed on the provided VM. The VM also provides a template of the spreadsheet, named `2024-ecm2426-ca-<uuid>.ods` (the `<uuid>` part will be replaced by a unique identifier) in the home directory of the user `lh` in a folder `coursework`, i.e., `/home/lh/coursework`.

On sheet one of the spreadsheet: For each answer, you will need to fill out a clearly marked field in the **column B** of the provided spreadsheet. If you want to declare the use of specific references, you can do this in **column D**. **Do not change the UUID in the last row of the spreadsheet.**

On sheet two of the spreadsheet: You will need to complete the declaration of the use of GenAI tools. This includes the declaration of the use of AI Tools (or the declaration that you have not used any AI tools). If you have used AI Tools, you will need to specify which AI tools you used and the column for naming references on sheet one should include a brief reflection on the use of AI for the question you used AI tools.

The VM contains a tool `check-submission` that you can use to check the validity of your spreasheet (in ODS format). **It is strongly recommended** that you use this tool to check the compliance of your spreadsheet to the required submission format, before submitting your solutions:

Bash

```
lh@ML-RefVm-326614:~$ check-submission 2024-ecm2426-ca-<uuid>.ods
```

This script will check the syntactic compliance of your submission, i.e., there is a solution for each question. It does not check for the correctness of your submission. Note that a successful run of the `check-submission` script does neither guarantee that your submission is complete nor that it follows the guidelines. *Checking that the submission is complete and that the individual entries comply to the specified format is your responsibility!*

You can copy the spreadsheet to your local machine using `scp` or `sftp`. Note that also for `scp` (and `sftp`), you need to configure the correct port (i.e., the same port used for the `ssh` command displayed in Azure Labs). Assuming that you can connect to the VM using the following command:

Bash

```
achim@logicalhacking:~$ ssh -p 65042 lh@host.azure.com
```

On your local machine (not the VM), you can use (note that `scp` uses an uppercase `P` for specifying the remote port):

Bash

```
achim@logicalhacking:~$
  scp -P 65042 lh@host.azure.com:coursework/2024-ecm2426-ca-<uuid>.ods .
```

to copy the spreadsheet to your local machine. Of course, you can also use a graphical front-end, e.g., putty or winscp.

Help, I Deleted a File

Note that it is your responsibility to not accidentally overwrite or delete any files. Still, some question might refer to files provided on the VM in a folder `/home/lh/coursework`. If you accidentally delete or modify such a file provided on the VM, you can recover it from a read-only backup available in the hidden directory `/home/lh/.coursework` (note that the directory starts with a '`.`' and, therefore is only shown when using the a parameter of `ls`, e.g., `ls -a`).

Questions

Question 1 ***Week 2: Foundations & Access Control*** **(20 marks)**

The virtual machine 2024-ecm2426-ca has several regular user accounts configured. In this exercise, you should explore these accounts and their access rights.

E.1.1. (5 marks): List all logins (not the full usernames) that both have numerical user id larger or equal than 2000 **and** that are a member of the group `staff`.
Submit your answer in cell B2, separate the logins with a comma, e.g., "`joe, jane, root`". Write "none", if there are no such users on your VM.

E.1.2. (5 marks): Analyse the VM to obtain the password of the user with the numerical user id 2000. Note that **you should obtain the password using reconnaissance**, not by running a computational expensive password cracking tool.
Submit your answer in cell B3 of the spreadsheet.

E.1.3. (5 marks): The user with the numerical user id 2001 has a file `flag1` in their home directory. What is the content of this file?
Submit your answer in cell B4 of the spreadsheet.

E.1.4. (5 marks): List *all* users with their login that can read the content of the file `flag2` in the home directory of the user with the numerical id 2001.
Submit your answer in cell B5 of the spreadsheet, separate the logins with a comma, e.g., "`joe, jane, root`". Write "none", if there are no users that can read the content of this file.

Question 2 ***Week 4: Cryptography, Signatures & PKIs*** **(15 marks)**

In the home directory of the user `lh` on the virtual machine 2024-ecm2426-ca threre is an X.509 certificate stored in a file called `/home/lh/coursework/exercise02/cert.pem`. This certificate has been created for securing a web server. The directory also contains three asymmetric key pairs, stored in the files `key0.pem`, `key1.pem`, and `key2.pem`. The directory also contains an encrypted data file (`secret.txt.enc`) and a digital signature `secret.enc.sha256.sig`. There is no password set on the key pairs.

For this exercise, you might want to use the `openssl` command line tool, which is provided on the virtual machine.

E.2.1. (2 marks): Name the signature algorithm that has been used for signing the certificate.
Submit your answer in cell B6 of the spreadsheet.

E.2.2. (3 marks): Is the certificate valid for the domain `www1.exeter.ac.uk`?
Submit your answer in cell B7 of the spreadsheet.

E.2.3. (3 marks): Is the certificate valid for the domain `www3.exeter.ac.uk`?
Submit your answer in cell B8 of the spreadsheet.

E.2.4 (2 marks): Name the key file (e.g., key1.pem) that contains the key used for creating the signature secret.enc.sha256.sig of the file secret.txt.enc.
Submit your answer in cell B9 of the spreadsheet. Write "none", if the signature has been created with a different key.

E.2.5. (5 marks): Decrypt the file secret.txt.enc. What it is content?
Submit your answer in cell B10 of the spreadsheet. Write "none", if the file has not been encrypted with any of the available key files.

Question 3 *Week 5: Security Protocols* (15 marks)

The provided VM allows you to access a network using that uses the IP range 172.17.0.*. Your virtual machine is connected to this network via the network interface docker0. This network contains a number of computers that might have insecure services installed.

As user lh, analyse the network traffic on the 172.17.0.* (e.g., using Wireshark) and answer the following questions related to the http traffic on this network.

E.3.1. (5 marks): Give the full URI of the password protected web area accessed in this session.
Submit your answer in cell B11 of the spreadsheet.

E.3.2. (5 marks): Give the username of the user who is successfully accessing the protected web area.
Submit your answer in cell B12 of the spreadsheet.

E.3.3. (5 marks): Give the (cleartext) password of the user accessing the protected area. Note that no computational expensive techniques (e.g., a brute-force attack on a password hash) are required.
Submit your answer in cell B13 of the spreadsheet.

Question 4 *Week 7: Formal Analysis of Security Protocols* (15 marks)

The file /home/lh/coursework/exercise04/ecm2426.AnB contains a (incomplete) security protocol specification in Alice&Bob notation. The specification is incomplete, as it lacks one fact of the initial knowledge (denoted by <fact>). Moreover, the last protocol step, that is required to complete the protocol securely, is missing. Answer the following questions with the help of ofmc.

E.4.1 (5 marks): One role has an incomplete initial knowledge, denoted by <fact> in the provided specification. Replace <fact> with the minimal fact that is required so that the protocol is executable.
Submit your answer in cell B14 of the spreadsheet.

E.4.2 (2 marks): Who needs to send the last message required to complete the protocol securely?
Submit your answer in cell B15 of the spreadsheet.

E.4.3 (3 marks): Who receives the last message required to complete the protocol securely?
Submit your answer in cell B16 of the spreadsheet.

E.4.4 (5 marks): What message needs to be exchanged in the last protocol step that is required to complete the protocol securely?
Submit your answer in cell B17 of the spreadsheet.

Question 5 *Week 9: (Manual) Dynamic Security Testing* (20 marks)

The VM has a small Message Board application installed that allows users to post (encrypted and unencrypted) messages. It can be accessed at `http://127.0.0.1:5000`, using a web browser running on the VM. The user `lh` can use `sudo` to start/stop the application:

Bash

```
lh@ML-RefVm-326614:~$ sudo systemctl start message-board
lh@ML-RefVm-326614:~$ sudo systemctl stop message-board
```

During the start of the application, the database is reset to an empty state. Note that also during a system restart, the database state is reset.

Assume you are a penetration tester tasked to analyse this application. You should test the application for Cross-Site Scripting (XSS) and SQL Injection (SQLi) vulnerabilities. In particular, you should find:

- One SQL Injection (SQLi) vulnerability that allows you to access a post that otherwise is not accessible.

  E5.1. (5 marks): What is the full URL used for exploiting the vulnerability? *Submit your answer in cell B18 of the spreadsheet.*

  E5.2. (5 marks): What is the content of this “secret” message?
  Submit your answer in cell B19 of the spreadsheet.

- One stored XSS vulnerability that allows you to access a session cookie.

  E5.3. (5 marks): What is name of the input field (as specified in the `name` attribute of the HTML `input` tag), in which the payload is injected? *Submit your answer in cell B20 of the spreadsheet.*

  E5.4. (5 marks): What is the content of the session cookie *on the site at which the payload is executed*?
  Submit your answer in cell B21 of the spreadsheet.

Question 6 *Week 10: Static Security Testing* (15 marks)

In the directory `/home/lh/coursework/exercise06/` there is copy of an earlier version of the message board application that you analysed in the last question.

Analyse the application using the static security scanner “bandit” (`https://github.com/PyCQA/bandit`). The bandit tool is part of the provided VM and the version installed on the VM will be used as a reference for assessing your solution.

You can run bandit as follows, and it should report three potential security issues:

Bash

```bash
achim@logicalhacking:~/coursework/exercise06/$ pipenv run bandit -r .
Test results:
>> Issue: [Bxxx]
>> Issue: [Bxxx]
>> Issue: [Bxxx]
```

The actual output will provide more details. **Not all of these findings are real vulnerabilities, nor is the classification of the type of vulnerability provided by bandit always correct**. For the purpose of the analysis, we limit use a very narrow scope and a very strict definition of vulnerability:

- We focus on software vulnerabilities in the application itself, i.e., its Python source code and templates used for generating HTML pages. We do not consider configuration issues (for example, using http instead of https or the fact that the application is configured to run in development mode, are out of scope for this exercise).
- We classify an issue as a vulnerability, only if it can actually be exploited by an attacker.

E6.1. (1 marks): What is the identifier (e.g., "B105") and line number/position (e.g, 42:17) of the first reported vulnerability? Separate both by a comma (e.g., B105, 42:17). *Submit your answer in cell B22 of the spreadsheet.*

E6.2. (4 marks): Can the first reported vulnerability be exploited by an attacker? If yes, what is the most specific CWE identifier for this vulnerability. *Submit your answer in cell B23 of the spreadsheet. If the vulnerability is not exploitable, write 'none', else specify the most specific CWE, e.g., 'CWE-89' for SQL Injection.*

E6.3. (1 marks): What is the identifier (e.g., B105) and line number/position (e.g, 42:17) of the second reported vulnerability? Separate both by a comma (e.g., B105, 42:17). *Submit your answer in cell B24 of the spreadsheet.*

E6.4. (4 marks): Can the second reported vulnerability be exploited by an attacker? If yes, what is the most specific CWE identifier for this vulnerability. *Submit your answer in cell B25 of the spreadsheet. If the vulnerability is not exploitable, write 'none', else specify the most specific CWE, e.g., 'CWE-89' for SQL Injection.*

E6.5. (1 marks): What is the identifier (e.g., B105) and line number/position (e.g, 42:17) of the third reported vulnerability? Separate both by a comma (e.g., B105, 42:17). *Submit your answer in cell B26 of the spreadsheet.*

E6.6. (4 marks): Can the third reported vulnerability be exploited by an attacker? If yes, what is the most specific CWE identifier for this vulnerability. *Submit your answer in cell B27 of the spreadsheet. If the vulnerability is not exploitable, write 'none', else specify the most specific CWE, e.g., 'CWE-89' for SQL Injection.*