NTNU

Department of (Computer and Information Science)

Examination paper for (TDT4237) (Software Security)

Academic contact during examination: Jingyue Li

Phone: 9189 7446

Examination date: 24-May-2018

Examination time (from-to): 15.00-17.00

Permitted examination support material: D

Other information:

Language: English

Number of pages (front page excluded): 7

Number of pages enclosed: 1

Informasjon om trykking av eksamensoppgave
Originalen er:
1-sidig X **2-sidig ☐**
sort/hvit ☐ **farger ☐**
skal ha flervalgskjema ☐

Checked by: Per Håkon Meland

_____

Date Signature

Students will find the examination results in Studentweb. Please contact the department if you have questions about your results. The Examinations Office will not be able to answer this.

Introduction

In this course, the written exam will count 70% of the final grade and the remaining 30% of the final grade comes from the compulsory exercises. So, your final grade of this course will be:
(Points you get from this written exam) * 70% + your grade of compulsory exercises.

If you feel that any of the problems require information that you do not find in the text, then you should

- Document the necessary assumptions
- Explain why you need them

Your answers should be brief and to the point.

Problem 1 – (40 points)

1) (5 points) Explain what heap overflow is, and list at least three methods/strategies to defend against heap overflow.
   Answer:

2) (5 points) Explain the Vigenère method to encrypt and decrypt string, and explain how to crack the Vigenère method.

3) (3 points) Explain how confidentiality and integrity are combined in SSL/TLS, IPSec, and SSH.

4) (2 points) Explain the SSL/TLS hand shake process.

5) (5 points) Explain the possible vulnerability of DAC (Discretionary access control), and explain why Bell-LaPadula model can help defend against the vulnerability.

6) (5 points) Explain what the “first one wins” principle of Android is and why such a principle can be vulnerable.

7) (5 points) Explain what web application firewall is in Azure, and explain why SQL injection, session fixation, session hijacking, and cross-site scripting can be fully or not fully defended by using the web application firewall.

8) (2 points) Explain what BSIMM (Building Security in Maturity Model) is, and propose how can a software company use such a model to improve security of own product.

9) (5 points) Explain what vulnerability the XML External Entities Attack exploits, how an attacker can exploit the vulnerability, and how to defend against such an attack.

10) (3 points) Explain what password salting is, what kinds of attack password salting can defend against and what kinds of attack it cannot defend against.

Problem 2 – (30 points in total)

For each of the code snippets listed below, your task is to:

- Identify all security vulnerability in the code (Note: you may find more than one vulnerabilities in one code snippet. You need to list and identify all of them.)
- Explain why these are security vulnerabilities/issues
- Fix the code (You may use pseudo-code for this. Remember to explain your solution).

Code snippet 1

Source: CWE-384

```
1.  <?php
2.  $SessionID = md5($UserName);
3.  if (empty($_COOKIE["SESSION_ID"]))
4.          setcookie("SESSION_ID",$SessionID);
5.  if ($_COOKIE["SESSION_ID"] == $SessionID):
6.          echo "Hello ".$UserName;
7.  else:
8.          echo "Please, enter your credentials";
9.  endif;
10. ?>
```

Code snippet 2

```
<form action="changeAddress.php" method="POST">
<p><input type="text" name="newAddress" /></p>
<p><input type="submit" value="Change Address" /></p>
</form>
```

changeAddress.php

```
<?php
    session_start();

    if (isset($_REQUEST['newAddress'])) {
      change_address($_REQUEST['newAddress']);
    }
    echo "<p>Your address has been changed to $newaddress </p>";
?>
```

Note: change_address () is a user defined function to store the new address into the database. We assume that this function is secure.

Code snippet 3

Source: CWE-613

```
<?php
if (empty($_COOKIE["SESSION_ID"])):
        $SessionID = GenerateSecureToken();
        setcookie("SESSION_ID",$SessionID, time()*3600);
elseif (ValidateSession($_COOKIE["SESSION_ID"])):
        echo "Hello ".$UserLogin;
else:
        echo "Please, enter your credentials";
endif;
?>
```

Code snippet 4

Source: https://www.acunetix.com/blog/articles

```
<html>
<head>
<title>Custom Dashboard </title>
    ...
</head> Main Dashboard for

<script>
```

```
8.     var pos=document.URL.indexOf("context=")+8;
9.     document.write(document.URL.substring(pos,document.URL.length));
10. </script>
11.     ...
12. </html>
```

Note: This is a web page http://www.example.com/userdashboard.html. The result of http://www.example.com/userdashboard.html?context=Mary would be a customized dashboard for Mary, containing the string "Main Dashboard for Mary" at the top.

Code snippet 5

Source: https://www.hackthis.co.uk

```
1.  <?php
2.    $page = $_GET;
3.    $filename = "/pages/$page";
4.    $file_handler = fopen($filename, "r");
5.    $contents = fread($file_handler, filesize($file));
6.    fclose($file_handler);
7.    echo   $contents;
8.  ?>
```

Problem 3 – (30 points)

Case description:

Company A is developing an IoT (Internet of Things) – based remote rehabilitation consulting service.

A patient with rehabilitation needs will log each day's activity using sensors installed within a wearable device. Using Bluetooth, the wearable device continuously communicates with an app of company A running on the patient's mobile phone (the mobile phone runs on Android platform). The activity data are stored at an external SD (Secure Digital) card of the patient's mobile phone. When the patient wants to upload the activity data to the web server of company A, the patient needs to log in the server first and then send the data. After the data is uploaded to the server, the corresponding data in the SD card will be deleted, to save space for new data.

When a therapist wants to read the activity data, the therapist needs also

to log in to the server. Based on some statistical analysis, the therapist can advise the patient to do certain exercises more often. The advice will be sent to the patient using emails. The patient pays the therapist based on advice provided by the therapist.
To use such a service, the patient needs to register his or her personal information, such as username, password, email address, age, gender, and some medical record to inform the therapist about the symptoms and what kinds of advice he or she needs. In addition, the patient can store credit card information in the server of company A for one-click payment. If the patient does not want to store the credit card information, the patient needs to type in such information every time he or she pays.
The therapist also need to register, and fill in some information, such as username, password, email address, name, office address, a short CV, and a bank account to receive the payment.

Your task is to make a risk-based assessment of this application based on RMF (Risk Management Framework).
Your tasks include:

- Identify business goals, business assets, and business risks (5 points).
- Identify at least 10 technical risks using threat modelling. The technical risks can be relevant to web server of company A and the mobile application of company A (10 points). (Note: You do not need to draw the threat modelling graphs. However, you need explain briefly how the threat modelling, e.g., misuse cases and attack trees, are applied to help you identify the technical risks.)

- Derive security requirements from each technical risk identified, and design and describe black-box penetration test cases (including test steps and expected results of each step) to verify each derived security requirement (10 points)

- This application must be compliant with General Data Protection Regulation(GDPR). List data of this application that can directly or indirectly identify a natural person, and discuss how company A

can address the privacy issue of this application from transparency, fair use, and minimalization perspectives (5 points)