**Title of Thesis**

**Possible Subtitle**

Author(s) First name Family name

Bachelor’s thesis or Master’s thesis

Month Year (e.g. September 2015)

Name of Field (e.g. Technology, communication and transport)

Name of Degree Programme (e.g. Degree Programme in Logistics Engineering)

|  |  |  |
| --- | --- | --- |
| jamk_fi_tunnus_sininen | | **Description** |
| Author(s)  Last name, First name | Type of publication  Master’s thesis | Date Month Year |
| Language of publication: |
| Number of pages | Permission for web publication: x |
| Title of publication  **Title**  Possible subtitle | | |
| Degree programme | | |
| Supervisor(s)  Last name, First name | | |
| Assigned by | | |
| Abstract  When completing this form, start from this field, on the row under the instructions, so that the font size remains 11.  The basic structure of the abstract is as follows:   * background * task and objectives * implementation method * results * conclusions.   In other words, the abstract summarises the work that has been done – not the content of the report. If there is room, the content of the report may be briefly mentioned.  The entire space reserved for the abstract must be used.  The abstract should be written in the past tense and with a passive voice. The text must not refer to the thesis, i.e., the words ‘this thesis’ must not be used. | | |
| Keywords/tags ([subjects](https://janet.finna.fi/Search/Results?lookfor=asiasanastot&prefiltered=format_Database&SearchForm_submit=Find&retainFilters=0&filter%5b%5d=format%3A%220%2FDatabase%2F%22&lng=en-gb)) | | |
| Miscellaneous ([Confidential information](https://intra.jamk.fi/opiskelijat/student/thesis/Pages/publicity.aspx)) | | |

YIIP3200 - YII14S1

Test Report for KIPITO OY

First Name - Last Name

STUDENT-ID: XXXX

Table of Contents

[1.0 SUMMARY 3](#_Toc411667621)

[1.1 Introduction 3](#_Toc411667622)

[1.2 Objective 3](#_Toc411667623)

[1.3 Requirements 3](#_Toc411667624)

[2. Sample Report –Summary 5](#_Toc411667625)

[2.1 Sample Report - Recommendations 5](#_Toc411667626)

[3. Sample Report – Technical Report 6](#_Toc411667627)

[3.1 Sample Report - Tools 6](#_Toc411667628)

[3.2 Sample Report - Executed Test Cases 6](#_Toc411667629)

[3.3 Sample Report – Information Gathering 6](#_Toc411667630)

[3.4 Sample Report – Service Enumeration 7](#_Toc411667631)

[3.5 Sample Report - Vulnerability Summary 8](#_Toc411667632)

[3.6 Sample Report - Vulnerability Details 9](#_Toc411667633)

[3.6.1 Ability Server 2.34 FTP STOR Buffer Overflow 9](#_Toc411667634)

[3.6.2 SQL Injection 10](#_Toc411667635)

[4. Sample Report - Attachments 11](#_Toc411667636)

# SUMMARY

## Introduction

This report will be graded from a standpoint of correctness and fullness to all aspects of the homework. The purpose of this report is to ensure that the student has a full understanding of security testing methodologies as well as the technical knowledge to meet the objectives of the course YIIP3200.

## Objective

The objective of this assessment is to perform an external (Homework1) and internal (Homework2) network security test against the KIPITO OY corporate network. In addition KIPITO OY’s outsourced WWW-site will be tested (Homework3). The student is tasked with following methodical approach in obtaining access to the objective goals. This test should simulate an actual network security test and how you would start from beginning to end, including the overall report. An example page has already been created for you at the latter portions of this document that should give you ample information on what is expected to pass this homework. Use the sample report as a guideline to get you through the reporting.

## Requirements

The student will be required to fill out this security testing report fully and to include the following sections:

* Overall High-Level Summary and Recommendations (non-technical)
  + Introduction
  + Scope of the test
  + How tests were conducted (project plan or similar)
  + Main findings
  + Recommendations
* Technical Report
  + Walkthrough and detailed outline of steps taken and tools used
* Each finding with included information; screenshots, walkthrough, and sample code if applicable.
* Any additional items that were not included

# Sample Report –Summary

This report contains results from the network security test towards KAPUTO OY’s public network. The focus of this test is to evaluate the network security, identify systems, and perform vulnerability analysis while reporting the findings back to KAPUTO OY.

How tests were conducted…

Main findings…

When performing the network security test, there were several alarming vulnerabilities that were identified on KAPUTO OY’s network.

## Sample Report - Recommendations

YII14S1 security tester recommends…

# Sample Report – Technical Report

YII14S1 security tester utilized a widely adopted approach to performing network security testing that is effective in testing how well the KAPUTO OY’s corporate networks are secured. Below is a breakout of how the testing was conducted and how the tester was able to identify the variety of systems and includes all individual vulnerabilities found.

## Sample Report - Tools

The following tools were used to conduct the security assessment.

|  |  |
| --- | --- |
| Tool and version | |
| Nmap | 6.40 |
| Nessus | 5.9 |

## Sample Report - Executed Test Cases

The following table contains a list of tests that were conducted during the test. This table does not contain any indication whether or not the test found any vulnerabilities. Detected vulnerabilities are listed in section XX.

|  |  |
| --- | --- |
| Test Cases | |
| Executed tests provided by Nmap | Port and service enumeration scan. |
| Executed tests provided by Nessus | Vulnerability scan. |

## Sample Report – Information Gathering

The information gathering portion of a test focuses on identifying the scope of the vulnerability assessment. During this test, YII14S1 security tester was tasked with KAPUTO OY’s public network. The specific IP addresses were:

**KAPUTO OY’s public IP-address**

192.168.1.2

## Sample Report – Service Enumeration

The service enumeration portion of a security test focuses on gathering information about what services are alive on a system or systems. This part provides detailed information on potential attack vectors into a system. Understanding what applications are running on the system gives the tester needed information before performing the actual vulnerability assessment.

|  |  |
| --- | --- |
| **Server IP Address** | **Ports Open** |
| 192.168.1.2 | **TCP:** 21,80,443  **UDP:** 161 |

## Sample Report - Vulnerability Summary

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Target\_name | | | | | |
| Test Information | | | | | |
| Start time: | | 13th February 2015 13:00 | | | |
| End time: | | 13th February 2014 20:00 | | | |
| Host Information | | | | | | |
| DNS Name: | | www.kaputo.fi | | | |
| IP: | | 192.168.1.2 | | | |
| OS: | | Linux Debian 6.0 | | | |
| Open Ports: | | tcp/21 (FTP)  tcp/80 (HTTP) nginx 1.1.19 | | | |
|  | | tcp/443 (HTTPS) nginx 1.1.19 | | | |
|  | | udp/161 | | | |
| Results Summary | | | | | | |
| High | | Medium | Low | Info | Total |
| 2 | | 0 | 0 | 0 | 2 |
| Vulnerablities | | | | | | |
| 3.6.1 Ability Server 2.34 FTP STOR Buffer Overflow | | | | | | |
| Synopsis: | The remote service is vulnerable to buffer overflow vulnerability | | | | | |
| 3.6.2 SQL Injection | | | | | | |
| Synopsis: | The remote host is vulnerable to SQL injection | | | | | |
| High vulnerability (Vulnerability Title) | | | | | | |
| Synopsis: | Synopsis text. | | | | | |
| Medium vulnerability (Vulnerability Title) | | | | | | |
| Synopsis: | Synopsis text | | | | | |
| Low vulnerability (Vulnerability Title) | | | | | | |
| Synopsis: | Synopsis text | | | | | |
| Info (Info Title) | | | | | | |
| Synopsis: | Synopsis text. | | | | | |

## Sample Report - Vulnerability Details

This section will give a more detailed information on each vulnerability. The details will cover explanation of the vulnerability, how it can be exploited if applicable, how the vulnerability can be mitigated or fixed, how severe the vulnerability is and additional information to help the reader to understand the vulnerability (e.q. screenshots, poc etc.).

|  |
| --- |
| Ability Server 2.34 FTP STOR Buffer Overflow **Synopsis:** The remote service is vulnerable to buffer overflow vulnerability  **Vulnerable Targets:** 192.168.1.2 tcp/21 (FTP)  **Vulnerability Explanation**: Ability Server 2.34 is subject to a buffer overflow vulnerability in STOR field. Attackers can use this vulnerability to cause arbitrary remote code execution and take completely control over the system.  **Vulnerability Fix**: The publishers of the Ability Server have issued a patch to fix this known issue. It can be found here: <http://www.code-crafters.com/abilityserver/>  **Severity: HIGH**  **References:** CVE-2014-0246  **CVSS Base Score:** X.X (CVSS2#AV:N/AC:M/Au:N/C:P/I:N/A:N)  **Proof of Concept Code Here:**  N/A **Screenshot Here:**  N/A |

|  |  |  |
| --- | --- | --- |
| **SQL Injection** **Synopsis:** The remote host is vulnerable to SQL injection  **Vulnerable Targets:**   |  | | --- | | 192.168.1.2/welcome.php tcp/80 | | 192.168.1.2/welcome.php tcp/443 |   **Vulnerability Explanation**: A custom web application identified was prone to SQL Injection attacks. When performing the web application test, YII14S1 security tester noticed error-based MySQL Injection on the hello query string parameter. While enumerating table data, it was possible to successfully extract login and password credentials that were unencrypted that also matched username and password accounts for the root user account on the operating system. This allows for a successful breach of the Linux-based operating system as well as all data contained on the system.  **Vulnerability Fix**: Since this is a custom web application, a specific update will not properly solve this issue. The application will need to be programmed to properly sanitize user-input data, ensure that the user is running off of a limited user account, and that any sensitive data stored within the SQL database is properly encrypted. Custom error messages are highly recommended, as it becomes more challenging for the attacker to exploit a given weakness if errors are not being presented back to them.  **Severity: HIGH**  **References:** CVE-2014-4444  **Proof of Concept Code Here:**  SELECT \* FROM login WHERE id = 1 or 1=1 AND user LIKE "%root%" **Screenshot Here:**  N/A |

# Sample Report - Attachments

Include your attachments to the document. Typically the report is kept separate from technical scanner data and the attachments portion of the report contains additional information about the vulnerabilities, used test methods or policy guidelines that were followed during the testing.