**Test and audit report**

**Lidl**

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)) | | |

Table of Contents

[1 Introduction 1](#_Toc501481998)

[2 Scope of the test 1](#_Toc501481999)

[3 Audit activities 1](#_Toc501482000)

[4 Main findings 1](#_Toc501482001)

[5 Recommendations 1](#_Toc501482002)

[6 Technical Report 1](#_Toc501482003)

[6.1 Tooling 2](#_Toc501482004)

[6.2 Executed Test Cases 2](#_Toc501482005)

[6.3 Information Gathering 2](#_Toc501482006)

[6.4 Service Enumeration 3](#_Toc501482007)

[6.5 Vulnerability Summary TO BE DONE 4](#_Toc501482008)

[6.6 Vulnerability Details TO BE DONE 5](#_Toc501482009)

[1.1.1 Ability Server 2.34 FTP STOR Buffer Overflow 5](#_Toc501482010)

[1.1.2 **SQL Injection** 6](#_Toc501482011)

[2. Sample Report - Attachments 7](#_Toc501482012)

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 LDIL corporate network. The auditor group is tasked with following methodical approach in obtaining access to the objective goals. This test simulates an actual network security test and how would one start from beginning to end, including the overall report.

Requirements

* 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

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…

TÄHÄN PÄÄTTYY ESIMERKKI RAPORTTI JA ALKAA MEIDÄN TUOTOS

# Introduction

This auditing report is a group exercise and it is part of the Auditing and Testing Technical Security course.

LDIL is a national e-tailing company that also has one physical retail store with a POS-system. LDIL business environment consists of information systems and different network domains. Target of this audit is LDIL's systems and networks.

# Scope of the audit

Everything?

Tätä rajattava…

# Audit activities

Three groups were created from group A auditing personnel and each group was assigned part of the LDIL network segments to perform the audit.

Groups were following:

* Vesa & Pinja
* Pauli, Jani, Otso & Janne
* Jouni, Teemu & Petri

# Main findings

Based on their auditing findings each subgroup presented main findings and summary of these findings is presented here.

# Recommendations

Fix update process to keep everything updated and fix firewall rules.

# Technical Report

YTC16S1 technical security testers are utilized to test and audit the LDIL corporate network thoroughly. Throughout the testing process, set of tools and preplanned test cases are planned - based on use cases given on LDIL business logic. The architecture of the LDIL is known as well as LDIL personnel are aware that audit and testing is performed to the corporate system. The testing type lies between crystal and grey-box combination - mainly testing the tester’s effectiveness and also the vulnerabilities on outdated system.

## Tooling

The following tools were used to conduct the security assessment. The tools are divided into information gathering, vulnerability scanning and also on web testing.

|  |  |
| --- | --- |
| Tool and version | |
| Nmap | 6.40 |
| Nessus | 5.9 |
| OpenVas |  |
| Burp Suite |  |

## 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.

Apart from the detailed test cases, also exploratory testing was applied by using Burp suite.

|  |  |
| --- | --- |
| Test Cases | |
| Executed tests provided by Nmap | Port and service enumeration scan. |
| Executed tests provided by Nessus | Vulnerability scan |
| Executed tests provided by OpenVas | Vulnerability scan |
| Burp Suite / Owasp ZAP | Web application testing, penetration testing |

## 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

## Service Enumeration

Liitteeseen tämä kappale ja tähän jokin teksti

Tähän kappaleeseen laitetaan segmenteittäin löydetyt avoimet portit.

* MGMT,
* Warehouse &
* ws/staff
* Internal
* Branch
* DMZ

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 |

## Vulnerability Summary Olisiko yhteenveto segmenteittäin?

DMZ taulukko

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Network segment | Critical | High | Medium | Low | Info |
| 10.0.0. |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 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. | | | | | |

## Vulnerability Details TO BE DONE

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** **q Here:**  N/A |

**Web-testing:**

### Outdated jQuery library in use

**Synopsis:** jQuery library is outdated and possibly vulnerable to exploits

**Vulnerable Targets:**

**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**: Update jQuery as well as the dependent libraries to the latest version.

**Severity: MEDIUM**

**References:** None

**Proof of Concept Code Here:**

N/A  
**Screenshot Here:**

N/A

### Outdated PHP version in use

**Synopsis:** PHP framework is outdated and possibly vulnerable to exploits

**Vulnerable Targets:**

**Vulnerability Explanation**: Intra-server is running unsupported PHP framework version, meaning there is no longer fixes and updates received from the PHP community.

**Vulnerability Fix**: Update PHP to at least to the one of the supported versions. Current version is not supported and might contain vulnerabilities as the support is no longer extended.

**Severity: MEDIUM**

**References:** None

**Proof of Concept Code Here:**

N/A  
**Screenshot Here:**

N/A

### Verbose information about system version available in http response

**Synopsis:** HTTP response includes information the operating system.

**Vulnerable Targets:** http://intra.ldil.de

**Vulnerability Explanation**: HTTP response gives out unneeded information to the end user and thus compromising the system security.

**Vulnerability Fix**: Hide the verbose response of currently used software versions from the http response.

**Severity: MEDIUM**

**References:** None

**Proof of Concept Code Here:**

N/A  
**Screenshot Here:**

N/A

### Verbose information about PHP and Apache version available in http response

**Synopsis:**

**Vulnerable Targets:** http://intra.ldil.de

**Vulnerability Explanation**:

**Vulnerability Fix**:

**Severity: MEDIUM**

**References:** None

**Proof of Concept Code Here:**

N/A  
**Screenshot Here:**

N/A

### XSS-protection is not enabled

**Synopsis:** Web browser XSS protection is not enabled or is disabled by the configuration of the X-XSS-Protection HTTP response header on the web sserver.

**Vulnerable Targets:** http://intra.ldil.de

**Vulnerability Explanation**: The X-XSS-Protection HTTP response header allows the web server to enable or disable the web browser´s XSS protection mechanisms. The following values would attempt to enable it: X-XSS-Protection: 1; mode=block.

The X-XSS-Protection HTTP response header is currently supported on IE, Chrome and Safari.

**Vulnerability Fix**: Ensure that the web browser´s XSS filter is enabled by setting the X-XSS-Protection HTTP response header to “1”.

**Severity: MINOR**

**References:** OWASP – Guidelines for setting security headers

**Proof of Concept Code Here:**

N/A  
**Screenshot Here:**

N/A

### Buffer overflow detected

**Synopsis:** Buffer overflow errors are happening when the overwriting of memory spaces of the background web process, which should never been modified intentionally or unintentionally. Overwriting values of the IP (instruction pointer), BP (base pointer) and other registers causes exceptions, segmentation faults and the other process errors to occur.

**Vulnerable Targets:** http://intra.ldil.de

**Vulnerability Explanation**: Potential buffer overflow detected. The script closed the connection and threw a 500 Internal Server Error.

**Vulnerability Fix**: Rewrite the background program using proper return length checking. This will require a recompile of the background executable.

**Severity: MEDIUM**

**References:** None

**Proof of Concept Code Here:**

GET

https://intra.ldil.de/wp-content/themes/twentysixteen?query=xlScCqlemqpPtXbFamPILdDaLkKPaUyLMWHUlAa......  
**Screenshot Here:**

N/A

### Directory browsing is enabled

**Synopsis:** Directory browsing is enabled and it is possible to view the directory listing

**Vulnerable Targets:** https://intra.ldil.de/wp-admin

**Vulnerability Explanation**: It is possible to view the directory listing. Directory listing may reveal hidden scripts, include files, backup source files which can be accessed to read sensitive information.

**Vulnerability Fix**: Disable directory browsing. If the directory browsing cannot be disabled because of some other service needing it, make sure the listed files do not include any risks.

**Severity: MEDIUM**

**References:** Apache httpd - Options

**Proof of Concept Code Here:**

N/A  
**Screenshot Here:**

N/A

### Format string error

**Synopsis**: A format string error occurs when the submitted data of an input string is evaluated as a command by the application.

**Vulnerable Targets:** https://intra.ldil.de

**Vulnerability Explanation**: Potential format string error occurred. The script closed the connection on a /%s

**Vulnerability Fix**: Review the background program using proper deletion of bad character strings (parameterize). This will require a recompile of the background executable.

**Severity: MEDIUM**

**References:** https://www.owasp.org/index.php/Format\_string\_attack

**Proof of Concept Code Here:**

SAMPLE QUERY HERE QUERY  
**Screenshot Here:**

N/A

### X-frame-options header not set

**Synopsis:** X-Frame-Options header is not included in the HTTP response

**Vulnerable Targets:** http://intra.ldil.de

**Vulnerability Explanation**: X-Frame-Options header should be included in the HTTP response to protect against ClickJacking attacks.

**Vulnerability Fix**: Most modern web browsers support the X-Frame-Options HTTP header. Ensure it is set on all web pages returned to your site.

**Severity: MEDIUM**

**References:** MSDN – Combating clickjacking with x-frame-options

**Proof of Concept Code Here:**

N/A  
**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.