# Project Management Plan

## Trusted Computing in the Implementation of Application Layer Protocols

Michiel Broekman

March 24, 2005

University of Oxford
Software Engineering Programme


University of Nijmegen
Security of Systems Group

This page is intentionally left blank.

# Contents

# 1 Introduction

This document describes the research project that will be performed at the Oxford University Computing Laboratory (OUCL). In the first chapter some information is given on the concept of application layer protocols. Then a short introduction to Trusted Computing Platforms is given to familiarize the reader with a concept that is still quite unknown. After that an abstraction from application layer protocols is chosen to enable an approach from a security perspective without the need to describe each protocol individually. In the second chapter the problem definition is formulated and an approach to solving it is described. Chapter three presents the planning information, however the exact details may be subject to change. The final chapter shows which people are involved in the project and provides some contact details.

## 1.1 Application layer protocols

From the perspective of the Open Systems Interconnection (OSI) model the application layer protocols are part of the top layer of the network stack. Examples of application layer protocols are SMTP, POP, HTTP, FTP and DNS.

There are lots of security issues related to application layer protocols. Various web servers, mail servers and other internet service software contain bugs that let remote users do things that are harmful. For example remote users can gain control of the machine and can do whatever they want. The exposure to these kind of threats can be minimized by running only the necessary software and getting the latest patches, and using software that has a good reputation [3]. However there remain a lot of security issues that cannot be easily dealt with. Therefore firewalls exist that try to cope with these kind of issues.

Application layer firewalls are hosts running proxy servers that do not permit direct traffic between networks and perform logging and auditing of traffic that is going through them. These proxy applications are software components that run on the firewall and are application specific which means that only recognized protocols can be examined.

It is difficult to distinguish between all possible application layer protocols. Therefore a higher level approach must be chosen to cover all protocols without having to deal with the protocols one by one.

## 1.2 Trusted Computing Platforms

The ability to protect a computing platform by using only software has some inherent weaknesses [1]. Security solutions based on software depend on the correct installation and execution, which can be easily affected by other software that has been executed on the same system. Even the most robust software does not have complete control over its own integrity. Malicious software can bypass the mechanisms of the OS and corrupt the behavior of the OS.

Experts in information security say that some security issues cannot be solved by software alone and therefore trusted hardware is required. For example, the increasing e-business activities on the internet demand more security than is given at this moment. This has led to the Trusted Computing Group (TCG, which was formerly called the Trusted Computing Platform Alliance) that designs the specifications for computing platforms that are supposed to create trust for software processes, based on some extra hardware within the platform.

Basically a Trusted Platform is a computing platform that makes use of a trusted component. This trusted component is in the form of built-in hardware and is the basis of trust for software processes. The security functions of the security hardware in a Trusted Platform *must* be trusted. The hardware is a root of trust and is able to measure both the hardware and the software environment of a specific system. If the software is decided to be trustworthy for some purpose, then all other security functions and software can operate as normal processes. These roots of trust are the core TCG capabilities.

TCG technology provides mechanisms that can be of great value in increasing trust. A behaviorial definition of trust clarifies how the word trust is used in this context: *an entity can be trusted if it always behaves in the expected manner for the intended purpose* [1].

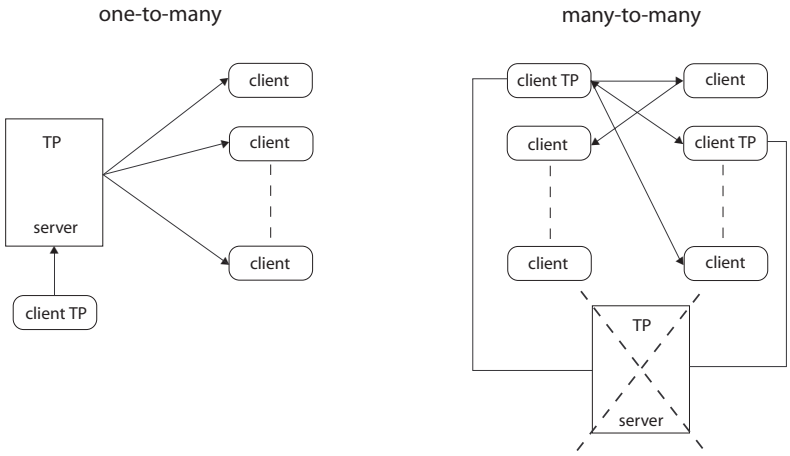
Trusted Computing makes it possible to improve security and robustness of distributed systems [4]. There are all kind of interesting applications that can be realized with TCG technology. One interesting application is that of distributed firewalls. On a Trusted Platform a distributed firewall is significantly more powerful. Another application is rate limiting for the prevention of distributed denial of service attacks (DDoS).

These kind of applications can only run on a Trusted Platform. There are a few papers that describe possible architectures. An interesting proposal about this can be read in [5].

## 1.3 Classification of application layer protocols

A somewhat higher level approach to application layer protocols can be realized by adopting another perspective. From the level of the application layer firewalls it is all about what the actual protocol messages look like. However, one can also look at how information is shared and disseminated. This perspective allows an abstraction from the application layer protocols and makes it easier to look at security issues from a more general point of view. At this stage only some basic information is given about the abstraction and there will be a further refinement in the master's thesis.

In the figure below a distinction is made between a one-to-many and a many-to-many connection to give some idea of how information can be shared and disseminated.




In the case of a one-to-many connection a server is connected to a number of clients. For example imagine the case where the server runs the Concurrent Versions System (CVS). The clients want to retrieve code that is saved on the server and must gain trust in the server in order to be sure that the code is not tampered with. This means that the server should be a Trusted Platform. A client that wants to update the CVS repository should be a Trusted Platform as well because the server needs to know for sure if the presented code is from a host that can be trusted.

Another case is that of the many-to-many connection. Here clients can connect to more than one client at the same time and share information. For example imagine the case of two clients that want to establish a connection in order to start a video-conference. These clients need to know if they can trust each other, because a malicious client can exploit the weaknesses of the

video-conference protocol. Therefore both clients should be Trusted Platforms. The server in the figure above is only used to setup a video-conference connection and does not play any further role in the data transfer between the two clients.

The classification outlined below is *inspired* by the Domain Based Security Model described in [2]. This model helps a defence-related organization to estimate security issues at different levels of details. The classification gives an abstraction from the protocols without focusing on implementation issues, as these are constantly subject to change. The following classification enables an interesting perspective on security issues.

**Message connection** is a connection where messages are sent from one place to another. As an example a short description is presented about how TCG technology can be used in the implementation of the protocols that are contained in this class. TCG technology can be used to limit the rate at which machines send emails, so that spam email can be controlled [4]. The implementation of a rate limiter with a Trusted Platform is not really difficult.

On every Trusted Platform a ticket-granting service should run in a closed-box virtual machine (VM). The ticket-granting service is allowed to release at most one ticket in a specific time period. The tickets depend on the content. In order to limit the rate at which a machine can send emails there first should be an open-box email client VM that tries to get a ticket from a ticket-granting VM for every email being sent. Before an email is sent the client VM sends a hash of the email to the ticket-granting VM, after which the resulting ticket is attached to the outgoing email. This results in that TC-enabled platforms in a network will not accept emails that contain no ticket or an invalid ticket. So at the end every machine can send at most one email every time period.

**Shared data repository connection** is a connection where a person from a particular domain (a logical place where a certain group of people use IT facilities in order to assist them in their business activities) publishes data so that others can look at it and possibly change it. The types of connections are listed below.

1. Filestore: this is a way of data sharing which uses a hierarchy of files and file containers. This structure enables tight control over the sharing of data.

2. Web: data is shared because it is published on a web server. The structures that are created may be more flexible, but the control over the sharing of data is a lot more difficult.

3. Database: data is only shared to people who have access to the database. The database structure is well defined, but implementing security requirements takes a lot of effort.

**Conferencing connection** is a connection that allows people to interact in a real-time manner. Although it gives people the real-time advantage it is really difficult to control the information that is sent. The types of connections are listed below.

1. Video-conferencing: this kind of connection is a client-to-client connection and requires a high bandwidth to transfer data. There is no intermediate server to avoid any delays in data transfer, although a server may be used for setting up a connection.

At this moment no further information about the classes will be provided and a definite classification will be presented in the master's thesis.

# 2 Research

The introductory information given in the previous chapter is meant to familiarize the reader with some basic concepts that will be further investigated during the research project. In this chapter the problem definition is described after which an approach and risk analysis are presented.

## 2.1 Problem definition

The research is focused around the general problem definition which can be formulated as follows:

*"What are the benefits and drawbacks of TCG technology in the implementation of application layer protocols?"*

TCG technology can be used in the implementation of application layer protocols. As already explained in the first chapter there are lots of security issues related to application layer protocols. During this research project there should be an investigation to what extend TCG technology can increase the security of application layer protocols. The benefits and drawbacks of implementing these security solutions should be considered and described in detail.

## 2.2 Approach

The main goal of this research project is to investigate what the benefits and drawbacks of TCG technology in the implementation of application layer protocols are. The approach used in this project can be divided in several phases:

- First of all Trusted Platforms and TCG technology should be described. It is important that the presented information is related to the rest of the research project, so too many details and off-topic subjects must be avoided. The reader should get a representative and unambiguous idea of what Trusted Platforms and TCG technology is all about.
- Next an abstraction from the application layer protocols should be presented. The link between the application layer protocols and the abstraction must be clarified. After that the security issues should be identified.
- Then TCG technology should be applied to the classes that were obtained in the previous phase. The extend to which TCG technology can be used to increase the security of these classes must be carefully

examined. This phase should also clarify what benefits can be found and what drawbacks must be faced.

- One or two relatively simple application layer protocols must be chosen. TCG technology can be applied to these candidates which demands a more low-level approach. Again, the benefits and drawbacks of using TCG technology in the implementation of these concrete protocols must be carefully investigated and described.
- At the end a reflection should be written and some possible future research in this area can be proposed.

### 2.3 Risk analysis

There are a few risks involved in this project that must be addressed. Each risk should be reduced to a minimum and therefore measures must be taken during the execution of this research project.

- It may be difficult to get a balance between the low-level details and the high-level details. To manage this risk it is of utmost importance to have regular meetings with Andrew Martin. Weekly meetings will probably suffice to minimize this risk.
- It may be tempting to talk in too general terms instead of really producing anything concrete. The structure of the master's thesis makes sure that the research progresses towards a continually more concrete idea. In the beginning the ideas may sound a bit vague but things will get more concrete while progress is made. Furthermore weekly meetings should minimize this risk.
- The chosen classification may not be sufficient to model the relevant characteristics. If this seems to be the problem there are two solutions: either another abstraction model should be chosen or the project should be adjusted.
- It may be difficult to match TCG technology with the classification. This risk is of course one of the exciting challenges of this research project and therefore it must be carefully examined.
- There is a danger that too many topics are focused on which results in an abundance of information and a lack of structure. It is of great importance to keep Andrew Martin posted as much as possible, so that he can intervene when necessary.
- If there is not enough time left then the last phase where TCG technology is applied to one or two candidate application layer protocols should be omitted or limited to one very simple protocol.

# 3 Planning

## 3.1 Timetable

The timetable spans a period of 6 months, the time in which the research project should be finished. The exact time of the periods may be subject to change.

| Task | Description | Time period |
|---|---|---|
| Project Management Plan | The main activity is to investigate what exact subject can be researched and what approach should be taken. This preliminary research will ultimately result in this project management plan. | to 22-03-2005 |
| Trusted Platforms and TCG technology | The relevant parts of Trusted Platforms and TCG technology should be investigated. The first chapter of the master's thesis deals with Trusted Platforms and TCG technology and should be finished at the end of this period. | to 18-04-2005 |
| Classification of application layer protocols | An abstraction from the application layer protocols should be given. The link between the application layer protocols and the abstraction (the classification that was outlined before must be further refined) should be clarified. Furthermore the security issues should be identified. This period will result in chapter two of the master's thesis. | to 18-05-2005 |
| TCG technology applied to classification | The extend to which TCG technology can be used to increase the security of these classes must be carefully examined. This phase should also clarify what benefits can be found and what drawbacks must be faced. This period will result in chapter three of the master's thesis. | to 20-06-2005 |
| TCG technology applied to a few application layer protocols | The end of the research project consists of applying TCG technology to one or two relatively simple protocols. More low-level details must be taken into account. Again, the benefits and drawbacks of using TCG technology in the implementation of these concrete protocols must be carefully investigated and described. This will result in chapter four of the master's thesis. | to 23-07-2005 |
| Writing master's thesis | The last parts of the master's thesis (reflection and future research) should be finished. It is important that all people who are involved in this project should be kept posted. Furthermore, feedback must be given as soon as possible to prevent unnecessary delays. At the end the master's thesis must be delivered. | to 31-07-2005 |
| Preparing presentation | The first presentation will be held at the OUCL and the final presentation will be held at the University of Nijmegen. | |

# 4 Project Organization

The research project will be carried out at the Oxford University Computing Laboratory (OUCL). There are two official supervisors involved in this project who will review my work. Andrew Martin, a university lecturer in software engineering, is my supervisor at the OUCL and Jaap-Henk Hoepman, associate professor at the SoS group, is my supervisor at the University of Nijmegen.

I will have weekly meetings with Andrew Martin and Andrew Cooper, a PhD student at the OUCL, to evaluate my progress. My contact with Jaap-Henk will be less intensive but important updates will be send to him on a regular basis. Most decisions are made with Andrew Martin, however major changes that will influence the research project are to be approved by Jaap-Henk. At all times both Andrew Martin and Jaap-Henk should agree with the research project. Jaap-Henk will judge the overall quality of the work in consultation with Andrew Martin.

## 4.1 Contact details

The people who are involved in this project can be reached at the email addresses indicated below.

| | |
|---|---|
| Michiel Broekman | Michiel.Broekman@comlab.ox.ac.uk |
| Andrew Martin | Andrew.Martin@comlab.ox.ac.uk |
| Andrew Cooper | Andrew.Cooper@comlab.ox.ac.uk |
| Jaap-Henk Hoepman | jhh@cs.ru.nl |

## References


[1] Boris Balacheff, Liqun Chen, Siani Pearson, David Plaquin, Graeme Proudler. *Trusted Computing Platforms, TCPA Technology in Context.*

[2] Chiew Peng Goh. *A Security Model For A Defence-Related Organization.*

[3] Paul D. Robertson, Matt Curtin, Marcus J. Ranum. *Internet Firewalls: Frequently Asked Questions.*

[4] Tal Garfinkel, Mendel Rosenblum, Dan Boneh. *Flexible OS Support and Applications for Trusted Computing.*

[5] Tal Garfinkel, Ben Pfaff, Jim Chow, Mendel Rosenblum, Dan Boneh. *Terra: A Virtual Machine-Based Platform for Trusted Computing.*