m-privacy GmbH
Werner-Voß-Damm 62
12101 Berlin

+49-30 24 34 23 34
FAX +49-30 992 968 56
info@m-privacy.de

Common Criteria: Security Target for the TightGate-Pro (CC) Version 1.4

Remote-Controlled Browser Systems Protection Profile (ReCoBS-PP)

Prepared by	m-privacy GmbH
Project Manager	Roman Maczkowsky
Project Document id	ST.odt
Version ID	Version 1.15
Date	2013-03-21

Green highlighted text refer to sections inside the PP sections, refined by the ST author

Table of Contents

1 SECURITY TARGET INTRODUCTION....4
1.1 Security Target Reference....4
1.2 TOE Overview....4
1.2.1 Overview....4
1.2.2 Usage and major security features....5
1.2.2.1 Idea and aim of the TOE....5
1.2.2.2 Intended environment....6
1.2.2.3 Basic description of the TOE functionality....7
1.2.2.4 TOE type....7
1.2.2.5 Required non TOE Hardware....8
1.3 TOE Description....8
1.3.1 Technical Overview of the TOE....9
1.3.2 TOE Components....9
1.3.3 Security Functionality offered by the TOE....10
2 CONFORMANCE CLAIMS....10
2.1 CC Conformance Claim....10
2.2 PP Claim....10
2.3 Conformance rationale....11
3 Security Problem Definition....11
3.1 Introduction....11
3.1.1 Assets....11
3.1.2 Subjects....11
3.2 Assumptions....12
3.3 Threats....13
3.4 Organisational Security Policies....14
4 SECURITY OBJECTIVES....14
4.1 SECURITY OBJECTIVES FOR THE TOE....14
4.2 Security Objectives for the Operational Environment....14
4.3 Security Objectives Rationale....16
4.3.1 Protection offered by the TOE against the Threats....17
4.3.2 Protection offered by the TOE environment against the Threats....20
4.3.3 Consideration of the assumptions....22
5 Extended Components Definition....24
5.1 Extended Components Rationale....24
6 SECURITY REQUIREMENTS....24
6.1 Security Functional Requirements for the TOE....24
6.1.1 Flow Control Policy “TOE transmission protocol”....24
6.1.2 FDP_IFC.1 Subset information flow control....25
6.1.3 FDP_IFF.1 Simple security attributes....25
6.1.4 FMT_MSA.1 Management of security attributes....25
6.1.5 FMT_MSA.3(t) Static attribute initialisation....26
6.1.6 FMT_SMF.1 Specification of Management Functions....26
6.1.7 FMT_SMR.1 Security roles....26
6.2 Security Assurance Requirements for the TOE....26
6.3 Security Requirements for the IT environment [removed]....28
6.4 Security Requirements Rationale....28
6.4.1 Security Functional Requirements Rationale....28

6.4.2 Dependency Rationale....30
7 TOE Summary Specification....33
7.1 TOE SECURITY FUNCTIONAL REQUIREMENTS....33
8 APPENDICES....34
8.1 I TERMINOLOGY....34
8.2 II ACRONYMS....35

1 SECURITY TARGET INTRODUCTION

1.1 Security Target Reference

ST Title	Common Criteria: Security Target for the TightGate-Pro (CC) Version 1.4 Remote-Controlled Browser Systems Protection Profile (ReCoBS-PP)
TOE	m-privacy TightGate-Pro (CC) Version 1.4
ST Version	1.15
Assurance level	EAL3+
Keywords	Firewall, RSBAC, RECOBS, Application Proxy, TightGate-Pro (CC) Version 1.4

1.2 TOE Overview

1.2.1 Overview

TightGate-Pro (CC) Version 1.4 is a workstation proxy firewall, which allows designated hosts to connect to, and use software on the server. By doing so, any potential threats are restricted to the TightGate-Pro (CC) Version 1.4 server, which is protected and designed to counter threats.
TightGate-Pro (CC) Version 1.4 is designed to be used to access all WWW and e-mail content, forwarding only graphical and textual information, or the result of any executable content, without actually forwarding the code required to generate a web page.
TightGate-Pro (CC) Version 1.4 is a Remote-Controlled Browsers System (ReCoBS) which is designed to be a modular part of a security gateway to enable almost unlimited access to content on the World Wide Web (WWW) or via e-mail from a Local Computer (LC) of a user inside a Local Network (LAN). At the same time it prevents both the local information of users as well as the local computer and net devices (machines) on the LAN from (negative) effects of malware contained in active content within web pages.

In brief, TightGate-Pro (CC) Version 1.4 is a ReCoBS which is intended for comfortable access to WWW and e-mail content on the Internet without compromising integrity, availability or confidentiality of information in the LAN:
• WWW and e-mail content can be accessed without severe restrictions (e.g. filtering of active content which severely limits the usability of some WWW content) – “access”
• Access occurs from the Local Computer (LC) of each user (i.e. no dedicated devices/networks for access necessary) – “comfortable”
• Access of WWW and e-mail content does not impair integrity, availability or confidentiality of information in the Local Network (LAN) – “secure”

Compared to other solutions for secure (in the sense of the definition above) WWW access the TOE does not require a dedicated and physically separated network or net devices but rather existing LCs and infrastructure can be reused (in combination with the TOE).

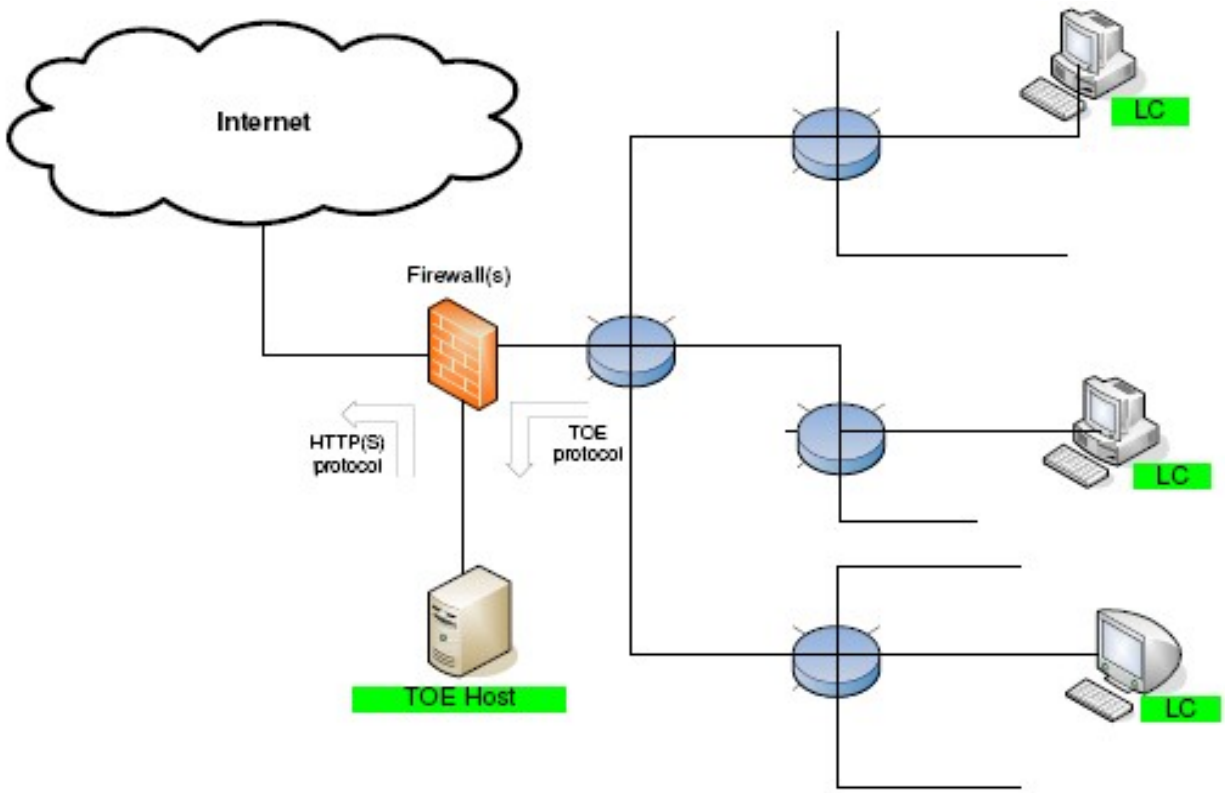

Figure 1

Figure 1: Schematic plot of a ReCoB system (running on systems marked in green). The TOE client is installed on the LC in the LAN, while the TOE server runs on a machine (called TOE host) in the DMZ, i.e. a machine which is separated from both the LAN as well as the Internet by firewalls.

1.2.2 Usage and major security features

1.2.2.1 Idea and aim of the TOE

TightGate-Pro (CC) Version 1.4 consists of an implementation of a TOE server, a TOE client and the TOE protocol as defined in appendix II. The TOE server which runs on one or more machines, referred to as the TOE host. TightGate-Pro (CC) Version 1.4 is not reliant on any particular hardware, and contains all the software required to execute the security functions while being designed to be situated in the Demilitarized Zone (DMZ) as part of the IT environment for the TOE. The TightGate-Pro (CC) Version 1.4 client runs on the LC. TightGate-Pro (CC) Version 1.4 server communicates with clients over a certain protocol referred to as the “TOE protocol”, which is part of the TOE. This protocol passes the firewall infrastructure (OE.Firewall) and traverses the network between the DMZ and the LC.

TightGate-Pro (CC) Version 1.4 is thus not a complete firewall and was not designed as such. TightGate-Pro (CC) Version 1.4 is one part of a complete security gateway for Internet access which has been designed to provide secure browsing and e-mail access while allowing integration into a firewall infrastructure.
The idea behind TightGate-Pro (CC) Version 1.4 is to intercept the information flow responsible for transforming HTML code (including active content) into pure audio-visual information. The increase of security is based on this interception and its inability to be bypassed. By separating the execution and the display environment the entire HTTP stream (i.e. HTML code, graphics, PDF files, etc.), including the problematic active content (like ActiveX controls, Java applets, JavaScript

programs), does not reach the LCs, only the comparatively harmless representation of this content as pure audio-visual data is transmitted onto the LCs.

To achieve this, the users run the TightGate-Pro (CC) Version 1.4 client on their LCs in the LAN, which connects to the TightGate-Pro (CC) Version 1.4 server which is a dedicated host situated in the DMZ. Each user is able to remotely control one (or more) browsers on the TightGate-Pro (CC) Version 1.4 server from his LC using the TOE protocol. The TOE protocol consists of key presses and mouse events (client to server), audio-visual data (server to client) and optionally limited clipboard exchange. As TightGate-Pro (CC) Version 1.4 and the browsers are contained in a single domain, all code embedded in the HTTP stream, including malware in active content, is executed within this sandbox.

Furthermore, access to WWW and optional e-mail content is granted only via TightGate-Pro (CC) Version 1.4. Hence possible side effects (both intended and unintended) are limited only to the TightGate-Pro (CC) Version 1.4 server and can not spread to other computers in the LAN. Since TightGate-Pro (CC) Version 1.4 fulfills dedicated security requirements the risk of a (temporarily accepted) compromise is greatly reduced. TightGate-Pro (CC) Version 1.4 implements the above functionality by using a specially tailored terminal server which protects against any untrusted code which may be running on the TightGate-Pro (CC) Version 1.4 server.

1.2.2.2 Intended environment

Typical environments for TightGate-Pro (CC) Version 1.4 are companies, (public) authorities or sections thereof where unlimited access to WWW content is required. TightGate-Pro (CC) Version 1.4 is intended to be part of an overall security infrastructure, like firewalls, intrusion detection systems, etc., which protects against threats from untrustworthy networks and data. TightGate-Pro (CC) Version 1.4 should not be used if – according to a risk analysis - a physically dedicated network with dedicated LCs solely for WWW access is required (e.g. because of highly sensitive or classified data in the LAN).

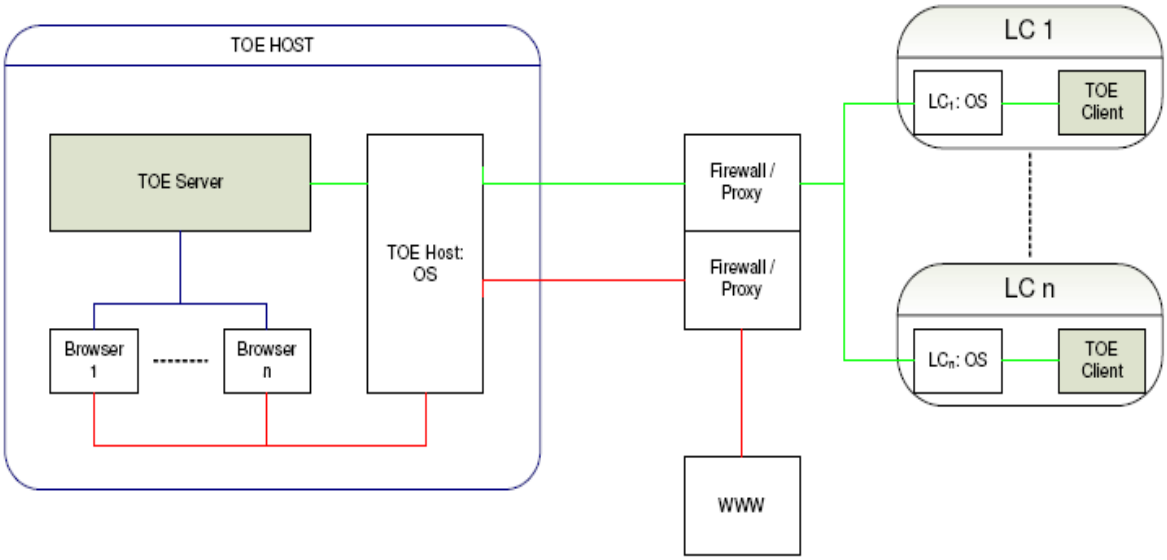

Figure 2

Figure 2: Schematic information flow of data from the WWW to the LC. TOE parts are denoted in

grey, the environment in white. The individual browsers communicate with the WWW using HTTP(S) (denoted in red), while the TOE server communicates with the TOE client using the TOE protocol (denoted in green). The information breach (denoted in blue) occurs on the TOE server. Only components relevant for the TOE are displayed, e.g. implementations will contain further devices (e.g. routers, switches).

1.2.2.3 Basic description of the TOE functionality

Embedded active content – along with all other content - can be used without limitations with TightGate-Pro (CC) Version 1.4. The representation of this content is then transmitted as pure (audio-)visual data via the TOE protocol to the client, where the graphical (and audio) representation of the content is displayed. Additionally TightGate-Pro (CC) Version 1.4 may offer the possibility for the user to copy a textual representation of the content from the TightGate-Pro (CC) Version 1.4 server to a clipboard on the LC. In converse, the user controls the browser remotely from his LC using the client. This control is achieved by transmitting key presses and mouse events from the client via the TOE protocol to the TightGate-Pro (CC) Version 1.4 server. Additionally TightGate-Pro (CC) Version 1.4 may offer the possibility for the user to paste textual content from a clipboard on the LC to the TOE server. Thus execution and display/control of (active) content are separated.

Integrity, availability and confidentiality of data in the LAN is ensured by TightGate-Pro (CC) Version 1.4 in conjunction with the firewall infrastructure, as the prevention of information processing by usage of the TOE protocol prevents any code (including malware) from the WWW executing on any computers in the LAN, and any data from the LAN reaching the browser (unless explicitly entered by the user).

The separation of information between the LAN and TightGate-Pro (CC) Version 1.4 includes identity information as well: to avoid that malware obtains identity information from TightGate-Pro (CC) Version 1.4 (e.g. valid combinations of usernames and passwords) and an attacker subsequently uses this identity information to open a direct connection inside the LAN (e.g. via remote login). TightGate-Pro (CC) Version 1.4 has a distinct identification and authentication system independent from any other authentication systems that may be used in the existing network (e.g. on the LC) and ensures that no (trivial) mapping of user attributes (like e-mail addresses) used on the TightGate-Pro (CC) Version 1.4 to those used inside the LAN is possible. Additional organizational measures have to be employed to avoid credentials (e.g. passwords) used on net devices from reuse with TightGate-Pro (CC) Version 1.4.

To allow for Copy and Paste TightGate-Pro (CC) Version 1.4 may offer the user the additional possibility to transfer pure text from WWW pages into the clipboard on the LC and individual textual contents of the clipboard on the LC after individual confirmation by the user as pure text to the TightGate-Pro (CC) Version 1.4 server. Both directions – if available - can be separately enabled by an administrator of TightGate-Pro (CC) Version 1.4 and default to off.

1.2.2.4 TOE type

Firewall component for secure WWW access

1.2.2.5 Required non TOE Hardware

The TightGate-Pro (CC) Version 1.4 server is designed to run on a hardware configuration existing of at a minimum:

Processor: 1.5GHz x86 compatible CPU(s)
RAM: 2GB
Storage: 20GB Free
Network: 100MBps

The TightGate-Pro (CC) Version 1.4 client will run on any computer running the Microsoft Windows XP or Windows 7 operating system.
The hardware required to run the TightGate-Pro (CC) Version 1.4 server is not included with TightGate-Pro (CC) Version 1.4, but has been verified to support TightGate-Pro (CC) Version 1.4. Untested hardware which is not necessary to meet the requirements of the TightGate-Pro (CC) Version 1.4 server is not supported for the configuration

1.3 TOE Description

TightGate-Pro (CC) Version 1.4 is the flagship component of the TightGate series of products. The TightGate-Pro (CC) Version 1.4 product comprises a firewall, hardened operating system, network services, and management system software. The firewall and its RSBAC hardened operating system are included in the scope of the TOE. All packages used for the TightGate-Pro (CC) Version 1.4 platform are based on the Debian or the Ubuntu Linux project, which both have an extensive integrity and quality assurance process. If any package has new vulnerabilities, an alert will be raised immediately, allowing appropriate measures to be taken.
TightGate-Pro (CC) Version 1.4 is a high availability firewall and workstation proxy for securing data communications and enabling continuous network connectivity. The services include e-mail, web browsing, and office suite helper applications. The RSBAC functionality is incorporated into the operating system. TightGate-Pro (CC) Version 1.4 is intended for use by organizations that need controlled, protected and audited access to services, both from inside and outside their organization's network, without risking a compromise of an internal network.
The TightGate-Pro (CC) Version 1.4 evaluated configuration provides:

- Access to the command line interface to the TOE Server from the operating system is disabled for administrators in the default operating mode, as the default command line interface is replaced by the menu system.
- Access to the WWW without any executable content relayed. Any executable content such as programs, scripts and documents have only their audio and visual representation transmitted to the TOE client, with any execution only able to occur on the TOE server. The audio visual representation is one way, directed from the TOE server to the TOE client, while the keyboard and mouse transmissions are also one way, and can only occur from the TOE client to the TOE server. There is the possibility to transmit textual clipboard data in one or both directions as configured by the administrator.
- The RSBAC policy parameters are configured to enforce policies and rules defined to ensure confidentiality, integrity and authenticity.
- Secure access to e-mail without any executable content relayed as defined for “Access to the WWW” above.

Auditing: TightGate-Pro (CC) Version 1.4 provides a means to generate audit records of security relevant events relating to the IP traffic through the firewall and RSBAC policy violation attempts. TightGate-Pro (CC) Version 1.4 provides a mechanism to prevent audit data loss.

Security Management and Protection of Security Functions: Administrators access the TOE host through the management server which provides the interface for managing the security policy and authentication attributes; the TSF data and security functions of the TOE server. TightGate-Pro (CC) Version 1.4 also ensures via the TOE host that RSBAC trusted security functions are always invoked and cannot be bypassed.

Data Protection: between the TightGate-Pro (CC) Version 1.4 server and any connecting LC's. TightGate-Pro (CC) Version 1.4 provides network security services and remote access based on the VNC over TLS protocol. This includes data confidentiality and integrity protection using TLS.

Self-testing: the TightGate-Pro (CC) Version 1.4 provides integrity checking, which can be triggered manually by an administrator only.
The integrity check is initiated from the read-only media, and mounts the target TOE host operating system disk as read-only. The check will verify unique crypto signatures of all the executable files, and all the static data. Signatures used for the integrity check are a tuple of MD5 and SHA256 security hashs. They are signed by a specific GnuPG key: 0xF033BC14 <repo@m-privacy.de>

1.3.1 Technical Overview of the TOE

TightGate-Pro (CC) Version 1.4 relies on RSBAC access control technology and the VNC protocol to control and provide connectivity and information flow between internal and external networks.

It also provides a means to keep the internal hosts IP-address private from external users. The TightGate-Pro (CC) Version 1.4 operating system and security services are based on m-privacys TightGate operating system, a modified Debian Linux distribution. As part of a solution, TightGate-Pro (CC) Version 1.4 provides high availability of firewall and application services for the users and ensures that users' machines will not become compromised.

TightGate-Pro (CC) Version 1.4 server is installed onto a computer meeting the minimum hardware requirements. After low level kernel initialization, RSBAC is enabled and the applied rules prevent any access not explicitly configured.

TightGate-Pro (CC) Version 1.4 server then starts an inetd daemon (Internet Daemon), which allows clients to connect to it. The client connection is handed over to a VNC server daemon, and the communication goes on by exchanging information over this connection via the TOE protocol. The TOE protocol is defined as the VNC protocol encapsulated in the TLS protocol. The TightGate-Pro (CC) Version 1.4 configuration of VNC requires TLS to be used to ensure confidentiality and accountability requirements.

1.3.2 TOE Components

TightGate-Pro (CC) Version 1.4 server requires a single or multi-processor(s) x86 of at least 1.5GHz. A hard-drive with a minimum of 20GB free space is required to install the required files and support a single user. The space required for TightGate-Pro (CC) Version 1.4 server increases with the amount of users it is intended to support.

TightGate-Pro (CC) Version 1.4 comes with an extensive documentation package which is intended as instructional guidance for administrators, covering each of the administration functions in detail and providing advice for troubleshooting and unexpected situations. TightGate-Pro (CC) Version

1.4 includes a comprehensive menu system comprising of system administration functions, auditing options, system control options to support the management and operation of the firewall with each option having a help menu detailing the intended operation.

The TOE host operating system includes RSBAC which implements various security models.

1.3.3 Security Functionality offered by the TOE

TightGate-Pro (CC) Version 1.4 enforces its defined set of rules via the integrated security functionality of the TOE host, especially the RSBAC access control framework. The possibility to assign roles and types to subjects and objects respectively with the RC model allows for any file or process to be restricted with a very granular level of control. The JAIL model allows for a process to be placed in a sandbox, in which communication with anything outside of the sandbox is not allowed unless explicitly defined. The CAP module provides functionality to separate one user's view over processes from all other system and users processes. It is also used to control otherwise unchecked special access granted by the Linux kernel called “capabilities”, for example denying root the traditional complete file system access (called “Discretionary Access Control Override” by the kernel). These additional special accesses are denied by default by the CAP module. The JAIL and RC models are used to prevent any executable content from being able to access the local network. Processes will be prevented from accessing and creating network sockets and from being available to any processes capable of transmitting to the local network.

To support the operations of TightGate-Pro (CC) Version 1.4, the management system includes a comprehensive menu system, that provides a trusted interface for administrator functions and a logging facility to store and manage (i.e., filter, sort, archive) the log records.
Its distributed architecture makes it flexible and scalable since it can run on a single or on multiple platforms.

2 CONFORMANCE CLAIMS

2.1 CC Conformance Claim

This TOE conforms to the following CC specifications:
Common Criteria for Information Technology Security Evaluation, Version 3.1 Revision 3 July 2009, ISO/IEC 15408:

- Part 2 conformant
- Part 3 conformant
- Package conformance to EAL3 augmented with ALC_CMS.4 and ALC_FLR.3.

2.2 PP Claim

This security target is conform against the remote controlled browsers systems protection profile ReCoBS-PP BSI-PP-0040 version 1.0 (2008-02-26) with strict-conformance.

2.3 Conformance rationale

This ST claims conformance to CC part 1, 2 and 3. As no SFRs or SARs were added this ST is in conformance with CC part 2 and 3. Further this ST claims conformance to the package EAL3 augmented with ALC_CMS.4 and ALC_FLR.3. As EAL3, ALC_CMS.4 and ALC_FLR.3 do not contain any uncompleted operations and both ALC_CMS.4 and ALC_FLR.3 do not contain any dependencies this conformance is satisfied (cf. also Section 6.2).
This security target for the TOE is directly in conformance with all sections of the common criteria. No augmentations or alterations other than those specified in the ReCoBs-PP were necessary.

3 Security Problem Definition

3.1 Introduction

3.1.1 Assets

The assets can be distinguished into primary and secondary assets. The main aim of this TOE is to protect the primary assets against manipulation and eavesdropping, as well as to avoid Denial of Service (DOS) attacks on them. The primary assets are:

- Data stored on machines in the LAN
- Data or information stored or transported in proximity to machines or devices in the LAN, e.g. printed information within the range of a camera connected to a machine in the LAN, spoken words within the range of a microphone connected to a net device in the LAN

Secondary assets are themselves of no value, but the possession or control of these assets enables or eases access to primary assets. Therefore these assets need to be protected as well.

- Credentials (i.e. authentication attributes like passwords) used on the LAN and TOE
- Security attributes (e.g. access permissions) on the TOE

3.1.2 Subjects

Administrator: A person who administers the TOE host and who is able to access the TOE on a dedicated service interface to:
1. add/remove users on the TOE
2. change security attributes of the **TOE Security Functions (TSF)**

User: A person at an LC authorized by an administrator to access the WWW via the TOE. This implies that the user is able to use the TOE client to connect to the TOE server ("initiate a session") and use a WWW browser to connect to WWW sites. He is further able to configure and use plugins required for display of special content and to interact with active content.

Attacker: A person who is neither a user nor an administrator and has no physical access to any net device in the LAN or DMZ, i.e. he can only attempt to access the TOE or net device on the LAN or

in the DMZ from outside the LAN and DMZ (typically from a machine on the internet).

The attacker is able to manipulate certain WWW sites in any way desired by him and to place any programs, including malware, on these sites, e.g. as active content in WWW sites. Those programs may be taken from public sources (e.g. sites on the WWW) and might include source code or may be developed specifically by the attacker for this attack. He is also able to spoof other electronic communication media like e-mail when in contact with users or administrators. The attacker is not able to physically access any net device in the LAN or the DMZ. The aim of the attack could be eavesdropping of (sensitive) information, manipulation of data (including configuration) on net devices inside the LAN or preventing access to data and services in the LAN. The attacker might be motivated financially or ideationally.

3.2 Assumptions

A.Firewall[1]: The TOE client runs on LCs inside the LAN. The TOE host is located in the DMZ and the TOE server runs thereon. Both the connection between the TOE host (situated in the DMZ) and any net device in the LAN as well as the connection between the TOE host and the Internet are separated by a firewall (cf. Figure 1). This firewall operates both on incoming as well as on outgoing traffic and includes network proxies, which operate on the transport (e.g. Internet Protocol) and additionally on the application layer for HTTP(S). The following rules are enforced by the firewall:
1. Connections from net devices in the LAN (including the LCs) to the TOE host can only use a port dedicated for the operation of the TOE server.
2. Only TOE clients running on net devices in the LAN can open connections to the TOE host.
3. It is impossible for the TOE host to initiate connections to net devices inside the LAN.
4. It is impossible for net devices inside the LAN to connect to content on the WWW directly, i.e. bypassing the TOE.
5. Only the TOE protocol as defined in paragraph 79 and 82 of BSI-PP-0040 can be used in connections between the TOE host and a TOE client in the LAN.

A.LC: The TOE clients - running on LCs inside the LAN - will not be manipulated by users or software on the LC.

A.Admin: The administrator is trustworthy, proficient and does not use this role to access WWW content.

A.Authentication: The users do not reuse any identification and authentication attribute (credential) on the TOE which has been used on any net device within the LAN.

Since the TOE server is intended to support simultaneous sessions of multiple users it offers an exposed target for sniffing data transmitted to and from the internet. Further any additional software could be used to attempt to bypass security functionality on the TOE host. To reduce the number of possible vulnerabilities in the IT environment, i.e. the number of flaws in the IT environment on the TOE host which could be exploited, only programs required for the TOE host and TOE server for proper operation are assumed to be installed:

A.Minimal: Only programs required for the operation of the TOE are available on the TOE host (e.g. TOE server, web browser, web browser extensions).

1 The term “firewall” is used here rather generically in the sense of a security gateway, cf. the definition in Sec. 8.

3.3 Threats

The TOE protects against threats originating from attackers as defined in Section 3.1.
For the purpose of the ST it is not relevant how malware is transported on the TOE host. In the following list of threats the malware is therefore assumed to be already on the TOE host. For this PP it is also irrelevant whether the malware runs completely autonomously on the TOE host or whether it is controlled remotely by the attacker.

T.Malware: A user downloads and opens/executes data (e.g. programs) from WWW sites (either explicitly or embedded in active content) onto an LC which impairs integrity, availability or confidentiality of data on net devices within the LAN.

Application note: A typical example would be a virus, which would transmit sensitive information (e.g. passwords) from the LC to sites outside the LAN.

T.Eavesdrop: Malware uses available hardware (e.g. web cameras, microphones) to eavesdrop the physical workplace of the user, i.e. the physical environment of the LC.

The following threats are a consequence of the design of the TOE. The TOE and TOE host provide additional security functionality to counteract or minimise these threats.

T.Credentials: An attacker deploys the authentication credentials obtained ("sniffed") on the TOE server to log onto a net device in the LAN directly.

Application note: An attacker could use credentials obtained on the TOE server, e.g. by exploiting a newly found vulnerability, to use a remote login facility for the LAN to connect to a net device in the LAN and hence completely bypass the TOE.

T.Hostcontrol: Malware running on the TOE host manipulates TSF data of either the TOE or the TOE host.

T.Hostcrossing: Malware running on the TOE host in the session of user A obtains or manipulates data belonging to a session of user B (where user B is an arbitrary user of the TOE different from A), denies B access to her data or runs malware (possibly including a copy of itself) within the session of B.

T.Spread: Malware running on the TOE host connects to an arbitrary net device in the LAN and transports another malware (or a copy of itself) on this net device, deploys this connection to obtain (sensitive) information from a net device in the LAN, manipulates information stored or processed on this net device or reduces the availability of net devices.

T.Clientspread: Malware running on the TOE host uses the TOE protocol to deploy this connection to obtain (sensitive) information from this LC, to manipulate information stored or processed on this LC, to reduce the availability of this LC or to transport some malware (e.g. a copy of itself) on the LC where the TOE client runs on.

3.4 Organisational Security Policies

Since the entire motivation for IT security functionality is based on countermeasures against explicitly listed threats no OSPs are defined.

4 SECURITY OBJECTIVES

The security objectives are an implementation independent description how the TOE counteracts the aforementioned threats. Also for each assumption regarding TOE usage the associated security objectives are described.

4.1 SECURITY OBJECTIVES FOR THE TOE

O.ServerToClient: The TOE server only transmits audio-visual data (i.e. graphical and audible representation of web pages) and those information necessary to present this data (like window size, placement requirements). The TOE server may additionally be able to transmit plain text marked on the TOE host by the user to the TOE client; if present this functionality has default to off and may only be activated by an administrator. The TOE client can only receive and present this kind of information; if text reception is activated it can only be sent into a clipboard on the LC.

O.ClientToServer: The TOE client can only transmit key presses (i.e. keyboard input) and mouse events explicitly directed towards the TOE client by the user, and the contents of a clearly designated configuration file at start up of the TOE client to the TOE server. The TOE server may additionally offer to transmit the plain text contents of a clipboard on the LC after explicit individual confirmation by the user for each transmission to the TOE server; if present this functionality has default to off and may only be activated by an administrator. The TOE client is unable to access any other data on the LC and transmit it to the TOE server.

4.2 Security Objectives for the Operational Environment

The following lists the security objectives for the environment:

OE.Firewall: The TOE client runs on LCs inside the LAN. The TOE host is located in the DMZ and the TOE server runs thereon. Both the connection between the TOE host (situated in the DMZ) and any net device in the LAN as well as the connection between the TOE host and the Internet are separated by a firewall (cf. Figure 1). This firewall operates both on incoming as well as on outgoing traffic and includes network proxies, which operate on the transport (e.g. Internet Protocol) and additionally on the application layer for HTTP(S). The following rules are enforced by the firewall:

1. Connections from net devices in the LAN (including the LCs) to the TOE host can only use a port dedicated for the operation of the TOE server.
2. Only TOE clients running on net devices in the LAN can open connections to the TOE host.
3. It is impossible for the TOE host to initiate connections to net devices inside the LAN.
4. It is impossible for net devices inside the LAN to connect to content on the WWW directly, i.e. bypassing the TOE.
5. Only the TOE protocol as defined in paragraph 4.1 and 4.1 can be used in connections between the TOE host and a TOE client in the LAN.

This prevents the bypass of the TOE and its functionality, e.g. malware running on the TOE host cannot connect to arbitrary net devices in the LAN or evade using the TOE protocol between the

TOE host and TOE client.

OE.LC: The TOE clients - running on LCs inside the LAN - will not be manipulated by users or software on the LC thus operates as specified in this PP.

OE.Admin: The administrator is trustworthy, proficient and does not use this role to access WWW content, since the TOE and its environment cannot defend themselves against misconfiguration or usage in administrator mode.

OE.Credentials: The TOE host operates an independent I&A system and offers the possibility to define rules for this system (e.g. specification of properties of the authentication attributes (credentials) used). Users do not use the same authentication attributes (credentials) for login on the TOE host as for any other net device in the LAN. This objective helps preventing transmission of passwords and login names used in the LAN into the potentially dangerous DMZ and attackers cannot use credentials obtained in the DMZ to connect directly to any net device within the LAN.

OE.Selfprotection: The IT environment (here: on the TOE host) prevents malware running on the TOE host during ordinary operation from manipulating TSF data of the TOE or the TOE host.

Application note: This and the following objectives for the environment might seem partially redundant. This is on purpose, as past experience has shown that systems running untrusted code experience new attack vectors (e.g. new types of vulnerabilities are found) and hence the "security" of such a system should not rely on a single objective.

OE.Manipulation: The IT environment (here: on the TOE host) ensures that no program running on the TOE host within a session of an user A (including, but not limited to, the browser, plugins/extensions and any active content) is able to access or manipulate data of an user B different from A, prevent B from accessing her data or runs malware within the session of B.

OE.Minimal: The IT environment (here: on the TOE host) ensures that only programs required for the operation of the TOE are available on the TOE host (e.g. TOE server, web browser, web browser extensions).

OE.Session: The IT environment (here: on the TOE host) ensures that:

1. At the beginning of the time of each session (i.e. after login on the TOE server), all programs accessible to the user on the TOE host are in a known state, i.e. executed on their own without any further user input[2], they only run code and access data as set up by an administrator. Especially it must be ensured that no application accesses any (active) content unknowingly by the user during initialisation (e.g. the address of the start page of the browser(s) cannot be altered by any program on the TOE host).
2. At the end of the time of each session (i.e. when logging off the TOE server), all programs running within this session (i.e. all programs part of this session) are terminated, including programs intended for later execution (if any). It must be ensured that no program is able to delay or continue execution beyond the end of the time of the session.

This objective ensures that malware is at most only active during the lifetime of a session and requires explicit user input to be loaded on the TOE (i.e. cannot be active at the start of the time of the session since the initial address for the WWW browser is fixed as well).

2 Application note: This could occur by entering a command name without parameters or by (double) clicking on an icon to start a program.

OE.Reset: The IT environment offers the facility to set up global time intervals where the entire TOE host including the TOE server is reinitialised to a known state, thereby terminating all sessions running at this point of time. The reinitialisation occurs in the following distinctive steps:

1. An integrity check on the entire static data of the mass storage on the TOE host is performed from outside the operating system normally running on the TOE host. Mass storage refers to the medium which stores the running operating system and all data related to the operating system and the TOE. Static data in this objective refers to all programs and (configuration) data which should not be altered during ordinary operation. During the integrity check it is important that the state of the mass storage cannot be altered.
2. The result of this integrity check is transferred to a device/account outside the TOE host[3] available to an administrator of the TOE.
3. The static data on the TOE host is reset to a known good state. It is important that only known dynamic files (for example user accounts) differ from the known good state (reference state). If the reset uses any file from the previously running TOE host, it has to be assured that these files are either identically to the files of the reference state (static data) or a detailed content analysis has to be performed so that no malware can be hidden within these files. The previously running operating system of the TOE host must not have access to the mass storage with the known good state.
4. The TOE host is booted from the mass storage that was reset to the known good state in step 3.

The manufacturer of the TOE host has to provide guidance how to handle the results of the integrity check, e.g. actions to take and if and how the manufacturer of the TOE host should be informed about failed integrity checks.
These regular reinitialisations enable an administrator to ensure that the entire TOE host and TOE server is in a known state (especially that no malware is running) at regular intervals (after which explicit user action is required to bring malware again on the TOE, cf. OE.Session).

4.3 Security Objectives Rationale

The following table provides an overview for security objectives coverage.

3 This could be a printer, a mobile phone, an e-mail account, but of course not a net device inside the LAN.

	O.ServerToClient	O.ClientToServer	OE.Firewall	OE.LC	OE.Admin	OE.Credentials	OE.Minimal	OE.Selfprotection	OE.Manipulation	OE.Session	OE.Reset
A.Firewall			X								
A.LC				X							
A.Admin					X						
A.Authentication						X					
A.Minimal							X				
T.Malware	X		X	X							
T.Eavesdrop	X	X	X	X							
T.Credentials						X					
T.Hostcontrol								X		X	X
T.Hostcrossing									X	X	X
T.Spread			X	X						X	X
T.Clientspread	X	X	X	X						X	

Table 1: Security Objective Rationale – mapping of threat and assumptions.

4.3.1 Protection offered by the TOE against the Threats

T.Malware: **A user downloads and open/executes data (e.g. programmes) from WWW sites (either explicityly or embedded in active content) onto an LC which impairs integrity, availability or confidentiality of data on net devices within the LAN.**

O.ServerToClient addresses this threat directly by strictly limiting the types of data transmitted from the WWW onto the LC to audio-visual data and placement information thus preventing arbitrary data including executable code to reach the LC via this channel (cf. also Footnote 50 for an additional discussion). Due to **OE.Firewall**, direct transmission of

arbitrary data from the WWW onto the LC (i.e. by using other channels and bypassing the TOE) is not possible either.

If the TOE offers Copy and Paste, a malware could in principle copy itself in a textual representation in the clipboard on the LC. Since the LC is not manipulated (**OE.LC**) all programmes on the LC are in the pre-configured state, especially the pasting has to be initiated by the user. Thus the attacker has to use another channel (e.g. social engineering) to convince the user to paste the textual representation of the malware into an appropriate interpreter (where an administrator of the LC has full control which interpreters are present at all). This is typically more difficult than sending the user a malware (e.g. contained in an document) and convince the user to open this document (and thus execute the malware). As enabling this limited Copy and Paste is, however, an additional risk, manufactures might omit this functionality and if it is present, the channel defaults to "off" and must be explicitly activated by an administrator (possibly following an individual risk analysis).

T.Eavesdrop: **Malware uses available hardware (e.g. web cameras, microphones) to eavesdrop the physical workplace of the user, i.e. the physical environement of the LC.**

O.ClientToServer addresses this threat directly by ensuring that only keyboard input ("key presses") and mouse events directed explicitly towards the TOE client, the contents of a clearly designated configuration file at start up (see next paragraph for rationale) and optionally individual paste actions (see below) will be transmitted to the TOE server (and hence further to WWW sites) thus preventing the contents of arbitrary files from being transferred and the transmission of arbitrary data streams (e.g. audio or video streams) onto the TOE host. Since the TOE client is not altered (manipulated), as ensured by **OE.LC**, no malware is present on the TOE which could simulate an input for the TOE client, the transport of such malware onto the client is prevented also (cf. **O.ServerToClient** and **OE.Firewall**). Furthermore the environment ensures that a direct connection from the TOE client to WWW sites, bypassing the TOE entirely, is not possible (cf. OE.Firewall) neither.

To allow some configuration to be stored on the non-manipulated LC (c.f. OE.LC and definition of attacker), e.g. credentials for the TOE host, the TOE client may send the contents of a clearly designated configuration file from the TOE client to the TOE server/host at startup. Later on (i.e. during operation) the TOE client cannot transmit any file contents to the TOE server thus again preventing the transmission of arbitrary file contents to the TOE server. Since the LC is not manipulated by users or software on the LC the file only contains the information as set up and not "arbitrary" information from the LC.

Finally the TOE may offer the possibility to transmit the contents of the clipboard of the LC to the TOE server. Since the LC is not manipulated, the transfer has to be initiated by the LC and each paste action has to be individually confirmed by the user, malware on the TOE host is unable to

access arbitrary content in the clipboard (no automatically shared clipboard is present!) but can possibly only access the textual data explicitly pasted by the user to the TOE server after explicit individual confirmation. As using the Copy and Paste channel is, however, an additional risk, manufactures might omit this functionality and if it is present, the channel defaults to "off" and must be explicitly activated by an administrator (possibly following an individual risk analysis).

T.Clientspread: **Malware running on the TOE host uses the TOE protocol to deploy this connection to obtain (sensitive) informtation from this LC, to manipulate information stored or processed on this LC, to reduce the availability of this LC or to transport some malware (e.g. a copy of itself) on the LC where the TOE client runs on.**

O.ServerToClient addresses this threat directly by strictly limiting the types of data transmitted from the WWW onto the LC to audio-visual data, formatting information and – optionally – textual information via Copy and Paste (see next paragraph) thus preventing arbitrary data including executable code to reach the LC via this channel[4]. Due to the objective **OE.Firewall** direct transmission of arbitrary data from the WWW onto the LC (i.e. by using other channels and bypassing the TOE) is not possible either.

If the TOE offers Copy and Paste, a malware could in principle use this channel to communicate with the TOE. Since the LC is not manipulated (**OE.LC**) all programmes on the LC are in the pre-configured state, especially the pasting of any textual information has to be initiated by the user. Thus the attacker has to user another channel (e.g. social engineering) to convince the user to paste the textual information into an appropriate interpreter (where an administrator of the LC has full control which interpreters are present at all) or any other place on the LC. Thus the user has full control over manipulation attempts of the malware. Similarly in the other direction malware on the TOE host can only access those information which has been explicitly confirmed by the user to be transferred from the LC to the TOE host, i.e. no shared clipboard allows "snooping" of information processed on the LC. As using Copy and Paste is, however, an additional risk, manufactures might omit this functionality (either or both transmission directions) and if it is present, the channels default to "off" and must be explicitly activated by an administrator (possibly following an individual risk analysis).

Since the TOE client is not manipulated (cf. OE.LC) it fulfils the client part of **O.ClientToServer** and will only transmit key presses and mouse events which have been directed explicitly towards the TOE client to the TOE server (and from there to sites in the WWW) and – optionally – individual contents of the clipboard (see previous paragraph for a

4 In principle any type of data can be encoded in the audio-visual data stream sent from the TOE server to the TOE client. Since the TOE client can only **display** the information sent and is not manipulated (OE.LC) no decoding of other information can occur on the LC, hence preventing arbitrary data (including commands) to be **received** by the LC via the TOE protocol.

discussion) and the TOE client will be unable to access arbitrary files/information on LC (the configuration file is the only file available for access to the TOE client, and this file can only be read before/while logging into the TOE host, when no malware is running within (part of) the session of the user logging in (cf. **OE.Session**)), i.e. only data (including clipboard contents) consciously sent by the user is received on the TOE host and malware is unable to secretly "pull" any other information (like, e.g., from a shared clipboard as used on some terminal systems not compliant to this PP). Finally the TOE protocol has a fixed maximum throughput from the TOE server to the TOE client (roughly the number of audio-visual information transmitted per time interval) which imposes a pre-defined maximum load onto the LC which has to display this information. Thus also the availability of the LC cannot be reduced by malware running on the TOE host.

4.3.2 Protection offered by the TOE environment against the Threats

T.Malware: **A user downloads and open/executes data (e.g. programmes) from WWW sites (either explicityly or embedded in active content) onto an LC which impairs integrity, availability or confidentiality of data on net devices within the LAN.**

OE.Firewall supports the countermeasures against this threat, as it prevents direct transmission of arbitrary data from the WWW onto the LC (i.e. by using other channels and bypassing the TOE). Thus all direct data transfer from WWW sites has to pass through the TOE (and especially through the TOE protocol which is decoded / encoded on the LC as defined in this PP (cf. **OE.LC**) and thus disallows a direct connection/channel to WWW sites) and therefor effectiveness of **O.ServerToClient** is ensured.

T.Eavesdrop: **Malware uses available hardware (e.g. web cameras, microphones) to eavesdrop the physical workplace of the user, i.e. the physical environement of the LC.**

OE.LC ensures that the TOE client is not altered (manipulated) and no malware is present on the LC which could simulate an input for the TOE client. Further the transport of such malware onto the client is prevented also (cf. **O.ServerToClient** which only allows pure audio-visual data and – if optionally pure text data – to be transmitted to the client (cf. also Footnote 50) and **OE.Firewall** which prevents further communication channels which could bypass the TOE). Finally this objective takes care that a direct connection from the LC to WWW sites, bypassing the TOE entirely, is not possible neither. These combined measures support **O.ClientToServer** and prevent the LC and its physical environment from being eavesdropped.

T.Credentials: **An attacker deploys the authentication credentials obtained ("sniffed") on the TOE server to log onto a net device in the LAN directly.**

OE.Credentials addresses this threat directly since it ensures that users have a credential for the TOE which is different from any credential used in the LAN. It also ensures that the TOE host provides an independent I&A system. Therefore there is no information on the TOE host which could be used by an attacker to log onto any net device within the LAN.

T.Hostcontrol: **Malware running on the TOE host manipulates TSF data of either the TOE or the TOE host.**

OE.Selfprotection addresses this threat directly since it ensures that malware running on the TOE host cannot manipulate TSF data of neither the TOE nor the TOE host. This objective is supported by **OE.Reset** which ensures that the TOE returns to a known state without any malware running and with well-known TSF data in regular time intervals and **OE.Session** which ensures that all programmes (including malware) running within (/are part of) a session are terminated[5] when the user logs off the TOE server. This objective reduces the impact of possible new vulnerabilities found after the certification of the TOE which might enable malware to (partially) bypass OE.Selfprotection as malware and (possible) modification of TSF data caused by it has a maximum lifetime on the system (and requires user action to return to the system afterwards). Finally during OE.Reset those modifications will be detected.

T.Hostcrossing: **Malware running on the TOE host in the session of user A obtains or manipulates data belonging to a session of user B (where user B is an arbitrary user of the TOE different from A), denies B access to her data or runs malware (possibily including a copy of itself) within the session of B.**

OE.Manipulation addresses this threat directly since it ensures that no programme (including malware) running on the TOE host within a session of user A is able to access or manipulate data of an user B different from A, to prevent B from accessing her data or executes malware within the session of user B.

This objective is supported by **OE.Session** which ensures that no malware is able to run at the start of the time of a session (e.g. when starting the browser) and all programmes run as part of a session (i.e. no programme with the rights of any user A runs after that user has logged off and before she has logged on again). Thus OE.Session reduces the impact of possible new vulnerabilities found after the certification of the TOE which might enable malware to (partially) bypass OE.Manipulation. Further malware has a maximum lifetime (ensured by **OE.Reset**) on the system (and requires user action to return to the system afterwards).

T.Spread: **Malware running on the TOE host connects to an arbitrary net device in the LAN and transport another malware (or a copy of itself) on this net device, deploys this connection to obtain (sensitive) information from a net device in the LAN, manipulates information stored or**

5 This only relates to programmes started by users, not to those associated to administrators or general tasks on the TOE host, cf. paragraph 4.2 and definition of "session" within this PP (in Chapter Fehler: Referenz nicht gefunden).

processed on this net device or reduces the availability of net devices.

OE.Firewall addresses this threat directly since it ensures that no programme (including malware) running on the TOE host is able to open a connection to any net device within the LAN. For countermeasures against attacks using existing connections, please see the countermeasures against T.Hostcrossing and T.Clientspread above. The reuse of existing connections to open new connections (from the TOE server to any net device) is not possible neither since the TOE client is unable to interpret such an request (it can only display audio visual data), cf. also Footnote 50 and will not be manipulated by users or software to do so (cf. **OE.LC**). This objective is supported by **OE.Session** which ensures that no malware is able to run when no user session is active (and idle users sessions, i.e. periods of time where the user is not interacting with the session (e.g. over night) are discouraged by the maximum lifetime of a session (cf. **OE.Reset**)). This objective reduces the impact of possible new communication paths through the firewall found after the certification of the TOE which might enable malware (limited) access to (information about) net devices within the LAN[6].

4.3.3 Consideration of the assumptions

A.Firewall: **The TOE client runs on LCs inside the LAN. The TOE host is located in the DMZ and the TOE server runs thereon. Both the connection between the TOE host (situated in the DMZ) and any net device in the lan as well as the connection between the OE host and the Internet are separated by a firewall (cf. Figure 1). This firewall operates both on incoming as well as on outgoing traffic and includes network proxies, which operate on the transport (e.g. Internet Protocol) and additionally on the application layer for HTTP(S). The following rules are enforced by the firewall:**

1. Connections from net devices in the LAN (including the LCs) to the TOE host can only use a port dedicated for the operation of the TOE server.
2. Only TOE clients running on net devices in the LAN can open connections to the TOE host.
3. It is impossible for the TOE host to initiate connections to net devices inside the LAN.
4. It is impossible for net devices inside the LAN to connect to content on the WWW directly, i.e. bypassing the TOE.
5. Only the TOE protocol as defined in paragraph 79 and 82 can be used in connections between the TOE host and a TOE client in the LAN.

OE.Firewall addresses this assumption directly as an requirement for the environment of the TOE.

A.LC **The TOE clients – running on LCs inside the LAN – will not be manipulated by users or software on the LC.**

OE.LC addresses this assumption directly as an requirement for the environment of the TOE.

6 Such a communication path might for example even be as limited as timing information gained from sending packages during periods of low network activity ("at night") to the firewall for obtaining information about the LAN.

A.Admin: **The administrator is trustworthy, proficient and does not use this role to access WWW content.**

OE.Admin addresses this assumption directly as an requirement for the administrator of the TOE. The competence of the administrator is important, since erroneous administration or usage with elevated (i.e. administrative) privileges has to be avoided.

A.Authentication: **The users do not reuse any identification and authentication attribute (credential) on the TOE which has been used on any net device within the LAN.**

OE.Credentials ensures the availability of an I&A system on the TOE host and addresses this assumption directly as an requirement for the users of the TOE.

A.Minimal: **Only programmes required for the operation of the TOE are available on the TOE host (e.g. TOE server, web browser, web browser extensions).**

OE.Minimal: addresses this assumption directly as an requirement for the environment (here: the TOE host) of the TOE. Since the TOE server is intended to support simultaneous sessions of multiple users running untrustworthy software (including malware) it offers an exposed target for monitoring or manipulating data transmitted to and from the internet. To reduce the number of possible exploitable vulnerabilities in the IT environment only programmes required for the TOE host and TOE server for proper operation are assumed to be present and hence able to access or operate on the TOE server.

As the TOE host consists of a complete operating system with graphical desktop, even the minimum amount of necessary software installed is quite huge.

A.Minimal therefore is interpreted for this TOE to only install the necessary software including helper apps, but no software for misappropriated tasks (e.g. network scanners).

The active content interpreted and executed by the web browser (including it's helper apps) as well as software-applications present on the TightGate-Pro (CC) Version 1.4 run by the end-users cannot manipulate, intercept, or eavedrop other TOE applications.

The browser on the TOE Host is running in it's unique RSBAC RC role and uses unique RSBAC RC types for storing it's data (including configuration data). Other RSBAC RC roles have no access to the web browser's data and processes, except for read/write access to the download area.

While the purpose of a ReCoB-System is to execute the dangerous tasks of webcontent interpretation on a secured and separated system, the end-users must be allowed to install web browser extensions and add-ons. On the TOE host those will always only been run within the RSBAC RC browser role.

Additional processes cannot alter the TOE protocol or configuration data, as the process handling and serving the TOE protocol is running as a different user and/or in it's own RSBAC RC role and types, which user and other processe's roles have no access to. The process serving the TOE

protocol is also responsible for setting the security attributes, which are also protected by the same RSBAC RC role and types.

5 Extended Components Definition

The ST does not contain any extended component.

5.1 Extended Components Rationale

As this ST does not contain any extended component no rationale is necessary.

6 SECURITY REQUIREMENTS

As stated in the CC, operations on the SFRs may be performed. Refinement and completion of operations are denoted by SMALL CAPS. If a refinement caused text to be replaced, the original version is printed stroken out. For the operations the uncompleted (original) version is given as a footnote. If the operation is to be completed by an author of an ST, the operation is enclosed in square brackets. Iterated components are denoted by a suffix to the component name, e.g. FMT_MSA.3**(h)**.

The refinements further highlighted in GREEN are operations which have been refined or completed by the ST author.

6.1 Security Functional Requirements for the TOE

6.1.1 Flow Control Policy "TOE transmission protocol"

Subjects: TOE server and TOE client
Objects: Any information exchanged between TOE server and TOE client
Security attributes: CopyPasteIn – Boolean value, default false
CopyPasteOut – Boolean value, default false
Management functions: SetCopyPasteIn – Allows to set security attribute "CopyPasteIn" on the TOE client
SetCopyPasteOut - Allows to set security attribute "CopyPasteOut" on the TOE client
Flow Control Policy:

1. The TOE server shall only send audio-visual data (graphics, sounds and data required for display (e.g. window size, placement information)) to the TOE client.
2. If CopyPasteIn is "true" then the TOE server shall also send plain text marked on the TOE host by the user to the TOE client which stores it in the clipboard on the LC.
3. The TOE client shall only send input events (i.e. key presses, mouse events) which have been explicitly directed towards the TOE client and (only during client startup) the contents of a clearly identified file to the TOE server.
4. If CopyPasteOut is "true" then the TOE client shall also send the textual contents of the clipboard on the LC to the TOE server after the user explicitly allowed this single transmission.

6.1.2 FDP_IFC.1 Subset information flow control

Hierarchical to: No other components.
Dependencies: FDP_IFF.1 Simple security attributes

FDP_IFC.1.1 The TSF shall enforce the TOE transmission protocol[7] on all information exchanged between TOE client and TOE server[8].

6.1.3 FDP_IFF.1 Simple security attributes

Hierarchical to: No other components.
Dependencies: FDP_IFC.1 Subset information flow control
FMT_MSA.3 Static attribute initialisation

FDP_IFF.1.1 The TSF shall enforce THE TOE TRANSMISSION PROTOCOL[9] based on the following types of subject and information security attributes: ALL INFORMATION EXCHANGED BETWEEN TOE CLIENT AND TOE SERVER AND THE SECURITY ATTRIBUTES COPYPASTEIN AND COPYPASTEOUT[10].

FDP_IFF.1.2 The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: THE INFORMATION EXCHANGE BETWEEN THE TOE CLIENT AND TOE SERVER OCCURS ACCORDING TO THE TOE TRANSMISSION PROTOCOL RULE 1 AND 3 INDEPENDENT OF ANY SECURITY ATTRIBUTE[11].

FDP_IFF.1.3 The TSF shall enforce ~~the~~ NO ADDITIONAL INFORMATION FLOW CONTROL SFP RULES[12].

FDP_IFF.1.4 The TSF shall explicitly authorise an information flow based on the following rules: THE INFORMATION EXCHANGE BETWEEN TOE CLIENT AND TOE SERVER CONFORMS TO THE TOE TRANSMISSION PROTOCOL RULE 4 (DEPENDING ON THE SETTING OF COPYPASTEOUT AND THE DECISION OF THE USER)[13].

FDP_IFF.1.5 The TSF shall explicitly deny an information flow based on the following rules: ANY INFORMATION EXCHANGE BETWEEN TOE CLIENT AND TOE SERVER NOT CONTAINED IN THE TOE TRANSMISSION PROTOCOL, INDEPENDENT OF ANY SECURITY ATTRIBUTE, TRANSMISSION OF PLAIN TEXT FROM THE TOE SERVER TO THE TOE CLIENT IF COPYPASTEIN IS FALSE AND TRANSMISSION OF PLAIN TEXT FROM THE TOE CLIENT TO THE TOE SERVER IF EITHER COPYPASTEOUT IS FALSE OR THE USER REJECTED THE TRANSMISSION[14].

6.1.4 FMT_MSA.1 Management of security attributes

Hierarchical to: No other components.
Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

FMT_MSA.1.1 The TSF shall enforce the TOE transmission protocol[15] to restrict the ability to change from restrictive to permissive[16] the security attributes COPYPASTEIN (IF

7 [assignment: information flow control SFP]
8 [assignment: list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects covered by the SFP]
9 [assignment: information flow control SFP]
10 [assignment: list of subjects and information controlled under the indicated SFP, and for each, the security attributes]
11 [assignment: for each operation, the security attribute-based relationship that must hold between subject and information security attributes]
12 [assignment: additional information flow control SFP rules]
13 [assignment: rules, based on security attributes, that explicitly authorise information flows]
14 [assignment: rules, based on security attributes, that explicitly deny information flows]
15 [assignment: access control SFP(s), information flow control SFP(s)]
16 [selection: change_default, query, modify, delete, [assignment: other operations]]

SETCOPYPASTEIN IS PART OF THE TOE, CF. FMT_SMF.1), COPYPASTEOUT (IF SETCOPYPASTEOUT IS PART OF THE TOE, CF. FMT_SMF.1)[17] to ADMINISTRATORS[18].

6.1.5 FMT_MSA.3(t) Static attribute initialisation

Hierarchical to: No other components.
Dependencies: FMT_MSA.1 Management of security attributes
FMT_SMR.1 Security roles

FMT_MSA.3.1(t) The TSF shall enforce the TOE TRANSMISSION PROTOCOL[19] to provide RESTRICTIVE[20] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2(t) The TSF shall allow the ADMINISTRATOR[21] to specify alternative initial values to override the default values when an object or information is created.

6.1.6 FMT_SMF.1 Specification of Management Functions

Hierarchical to: No other components.
Dependencies: No dependencies.

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions:SETCOPYPASTEIN, SETCOPYPASTEOUT.[22]

6.1.7 FMT_SMR.1 Security roles

Hierarchical to: No other components.
Dependencies: FIA_UID.1 Timing of identification

FMT_SMR.1.1 The TSF shall maintain the roles ADMINISTRATOR AND USER[23].

FMT_SMR.1.2 The TSF shall be able to associate users with roles.

6.2 Security Assurance Requirements for the TOE

The assurance requirements for the evaluation of the TOE, its development and operating environment are to be chosen as the predefined assurance package EAL3 augmented by the following components:

- ALC_CMS.4 (Problem tracking CM coverage), and
- ALC_FLR.3 (Systematic flaw remediation).

The resulting assurance package is represented below (the components augmented are printed in bold):

Assurance component, cf. CC part 3	Short description
ADV_ARC.1	Security architecture description
ADV_FSP.3	Functional specification with complete summary

17 [assignment: list of security attributes]
18 [assignment: the authorised identified roles]
19 [assignment: access control SFP, information flow control SFP]
20 [selection, choose one of: restrictive, permissive, [assignment: other property]]
21 [assignment: the authorised identified roles]
22 [assignment: list of management functions to be provided by the TSF]
23 [assignment: the authorised identified roles]

ADV_TDS.2	Architectural design
AGD_OPE.1	Operational user guidance
AGD_PRE.1	Preparative procedures
ALC_CMC.3	Authorisation controls
ALC_CMS.4	**Problem tracking CM coverage**
ALC_DEL.1	Delivery procedures
ALC_DVS.1	Identification of security measures
ALC_FLR.3	**Systematic flaw remediation**
ALC_LCD.1	Developer defined life-cycle model
ASE_CCL.1	Conformance claims
ASE_ECD.1	Extended components definition
ASE_INT.1	ST introduction
ASE_OBJ.2	Security objectives
ASE_REQ.2	Derived security requirements
ASE_SPD.1	Security problem definition
ASE_TSS.1	TOE summary specification
ATE_COV.2	Analysis of coverage
ATE_DPT.1	Testing: basic design
ATE_FUN.1	Functional testing
ATE_IND.2	Independent testing - sample
AVA_VAN.2	Vulnerability analysis

Table 2: Evaluation assurance package

EAL3 was chosen as it provides a good balance between assurance and economical feasibility of evaluations of TOEs in order to provide incentives for manufactures to evaluate their ReCoBS system against this PP.

EAL3 is augmented by ALC_CMS.4 as the TOE host is designed to run unknown, untrusted and malicious software (malware) and hence confidence must be established that the developer properly tracks and considers security flaws and their resolution during the development of the TOE in order to avoid known classes of vulnerabilities to affect the TOE even if the exact exploit might yet be unknown during the evaluation.

Further EAL3 is augmented by ALC_FLR.3 as during operations new vulnerabilities will be discovered by the organisations deploying the TOE. In order to deploy corrective actions, the developer has to provide the user with guidance how to report the security vulnerability and has to be able to accept, track and properly act on those reports. ALC_FLR.3 ensures that these guidance and procedures are available for the TOE.

6.3 Security Requirements for the IT environment [removed]

6.4 Security Requirements Rationale

6.4.1 Security Functional Requirements Rationale

The following table provides an overview for security functional requirements coverage. For convenience for developers (cf. paragraph 110 of BSI-PP-0040) hints for a rational for SFRs for the IT environment (for objectives for the IT environment where SFRs are sensible) are given as well.

	O.ServerToClient	O.ClientToServer	OE.Credentials	OE.Selfprotection	OE.Manipulation	OE.Session	OE.Reset
FDP_IFC.1	**X**	**X**					
FDP_IFF.1	**X**	**X**					
FMT_MSA.1	**X**	**X**					
FMT_MSA.3(t)	**X**	**X**					
FMT_SMF.1	**X**	**X**					
FMT_SMR.1	**X**	**X**					
FDP_ACC.2					**X**	**X**	
FDP_ACF.1					**X**	**X**	
FIA_SOS.1			**X**				
FIA_UAU.2			**X**		**X**		
FIA_UID.2			**X**		**X**		
FMT_MSA.3(h)				**X**	**X**		
FMT_SMR.2			**X**		**X**		
FPT_TST.1							**X**

Table 3: Security Objectives for the TOE by SFR

O.ServerToClient: The TOE server only transmits audio-visual data (i.e. graphical and audible representation of web pages) and those information necessary to present this data (like window size, placement requirements). The TOE server may additionally be able to transmit plain text marked on the TOE host by the user to the TOE client; if present this functionality has default to off and may only be activable by an administrator. The TOE client can only receive and present this kind of information; if text reception is activated it can only be sent into a clipboard on the LC.

FDP_IFC.1 and **FDP_IFF.1** ensure that all communication between TOE server and TOE client

obeys the TOE transmission protocol, which explicitly states that the TOE server may only transmit audio-visual data and formatting information required for display to the TOE client. If the ST author selected to include the possibility for Copying (i.e. to include SetCopyPasteIn in his ST) in FDP_IFF.1.4 and FMT_SMF.1 then the textual contents of paste actions into a clipboard on the LC are also transmitted from the TOE server to the TOE client. This additional transmission – if present - can only be activated by an administrator (**FMT_MSA.1** and **FMT_SMR.1**) and is restrictively configured (**FMT_MSA.3(t)**), i.e. textual transmission defaults to "off", allowing an administrator to perform a risk analysis before setting a new default value.

O.ClientToServer: The TOE client can only transmit:

- key presses (i.e. keyboard input) and mouse events explicitly directed towards the TOE client by the user, and
- the contents of a clearly designated configuration file at start-up of the TOE clientt

to the TOE server. The TOE server may additionally offer to transmit the plain text contents of a clipboard on the LC after explicit individual confirmation by the user for each transmission to the TOE server; if present this functionality has default to off and may only be activable by an administrator. The TOE client is unable to access any other data on the LC and transmit it to the TOE server.
FDP_IFC.1 and **FDP_IFF.1** ensure that all communication between TOE server and TOE client obeys the TOE transmission protocol, which explicitly states that the TOE client may only transmit key presses and mouse events which have been explicitly directed towards the TOE client, and (during start-up of the TOE client) the contents of a clearly marked file to the TOE server. If the ST author selected to include the possibility for controlled Pasting (i.e. to include SetCopyPasteOut in his ST) in FDP_IFF.1.4 and **FMT_SMF.1** then the textual contents of copy actions from the LC to the TOE server after individual confirmation by the user are also transmitted from the TOE client to the TOE server. This additional transmission – if present - can only be activated by an administrator (**FMT_MSA.1** and **FMT_SMR.1**) and is restrictively configured (**FMT_MSA.3(t)**), i.e. textual transmission defaults to "off", allowing an administrator to perform a risk analysis before setting a new default value.

6.4.2 Dependency Rationale

The following table provides an overview over all SFRs and their dependencies. For convenience for developers (cf. paragraph 110 of BSI-PP-0040) the suggested SFRs for the IT environment are included as well.

Reference	SFR	Dependencies	Comment
6.1.2	FDP_IFC.1 Subset information flow control	6.1.3 FDP_IFF.1 Simple security attributes	
6.1.3	FDP_IFF.1 Simple security attributes	6.1.2 FDP_IFC.1 Subset information flow control	

		6.1.5 FMT_MSA.3(t) Static attribute initialisation	
6.1.4	FMT_MSA.1 Management of security attributes	[FDP_ACC.1 Subset access control or 6.1.2 FDP_IFC.1 Subset information flow control] 6.1.7 FMT_SMR.1 Security roles 6.1.6 FMT_SMF.1 Specification of Management Functions	For the first dependency FDP_IFC.1 was chosen.
6.1.5	FMT_MSA.3(t) Static attribute initialization	6.1.4 FMT_MSA.1 Management of security attributes 6.1.7 FMT_SMR.1 Security roles	
6.1.6	FMT_SMF.1 Specification of management functions	none	
6.1.7	FMT_SMR.1 Security Roles	FIA_UID.1 Timing of identification	Dependency **not** satisfied, see rationale below.
6.3.3	FDP_ACC.2 *Complete access control*	6.2.4 FDP_ACF.1 Security attribute based access control	FDP.ACC.2 is hierarchical to FDP.ACC.1
6.3.4	FDP_ACF.1	6.2.3 FDP_ACC.2	

	Security attribute based access control	Complete access control 6.2.8 FMT_MSA.3(h) Static attribute initialization	
6.3.5	FIA_SOS.1 *Verification of secrets*	none	
6.3.6	FIA_UAU.2 *User authentication before any action*	FIA.UID.1 Timing of identification	Fulfilled by 6.2.7 FIA.UID.2 User identification before any action
6.3.7	FIA_UID.2 *User identification before any action*	none	FIA.UID.2 is hierarchical to FIA.UID.1
6.3.8	FMT_MSA.3(h) *Static attribute initialization*	FMT_MSA.1 Static attribute Management of security attributes 6.2.9 FMT_SMR.2 Restrictions on security roles	
6.3.9	FMT_SMR.2 *Restrictions on security roles*	6.2.7 FIA_UID.2 User identification before any action	FMT.SMR.2 is hierarchical to FMT.SMR.1
6.3.10	FPT_TST.1 *Testing*	none	

Table 4: Dependencies between the SFR for the TOE

As shown in Table 4 all dependencies (except for 6.1.7 - FMT_SMR.1 Security roles see below) are fulfilled either directly or by functional requirements hierarchical to the dependency.

FMT_SMR.1 depends on FIA_UID.1 which is not fulfilled in this ST Since the entire identification and authentication (I&A) is provided by the TOE host (i.e. the IT environment) it is not possible for the TOE to enforce the timing of the identification but rather it has to assume that the IT environment only allows access after identification (and authentication).

7 TOE Summary Specification

7.1 TOE SECURITY FUNCTIONAL REQUIREMENTS

The Target of Evaluation (TOE) consists of the TightGate-Pro (CC) Version 1.4 product including the TightGate-Pro (CC) Version 1.4 server and the TightGate-Pro (CC) Version 1.4 client. It provides the following implemented security requirements

Reference	SFR	Implementation
6.1.2	FDP_IFC.1 Subset information flow control	This is implemented by the XVNC Server which is supplying the TOE protocol and enforce the settings and the use of the TOE protocol. RSBAC RC provides the unbypassability.
6.1.3	FDP_IFF.1 Simple security attributes	This is implemented by the XVNC Server which is supplying the TOE protocol and enforce the settings and the use of the TOE protocol. RSBAC RC provides the unbypassability. The XVNC Server, by relaying the TOE protocol, transmit a visual representation of the WWW content. IT provides clipboard transfer according to the values of SECURITY ATTRIBUTES COPYPASTEIN AND COPYPASTEOUT set by config administrator role. The audio content is relayed via the PulseAudio sound daemon to a proxy server (not hosted on the TOE host).The proxy server relays the audio to the LC.
6.1.4	FMT_MSA.1 Management of security attributes	The restrictive default values “off” for COPYPASTEIN and COPYPASTEOUT are installed with the mprivacy-tools-CC packet. The restrictive default value can only be changed from the menu system for the config administrator account.
6.1.5	FMT_MSA.3(t) Static attribute initialization	This is implemented by the TightGate-Pro (CC) Version 1.4 RSBAC-configuration (provided via rsbac-policy-tgpro-CC packet). It allows only the VNC-Service to be accessed from the defined client IP-Range. The clients IP-range must be set by the config administrator (config menu).

6.1.6	FMT_SMF.1 Specification of Management	This is implemented by the config administrator menu system (config menu).
6.1.7	FMT_SMR.1 Security Roles	This is implemented by predefined, task specific ADMINISTRATOR roles with separated duties (maint, config, backuser, update). The association of roles is done at organisational level and no user role can switch to any administrative role on the TOE host.

Table 5: Implementation of SFR's in the TightGate-Pro (CC) Version 1.4

8 APPENDICES

8.1 I TERMINOLOGY

Active content – A program which is integrated in a web page and delivered to the browser upon accessing that web page for executing (on the TOE host). Examples are ActiveX and JavaScript.
Authentication attribute – A means to demonstrate the presence of a certain person. A typical example possible for the TOE are passwords. Synonymous to credential in this PP.
Configuration data – Variable data which is required to ensure the intended operation of the TOE and its environment, e.g. access rights and passwords.
Credential – see authentication attribute
Demilitarised Zone – (Part of) a network which is separated both from the LAN as well as from the Internet by firewalls.
Firewall – A system of hard or software based components which ensure secure linkage of IP networks by limiting the technically possible communication to those defined in a security policy. Sometimes the term “security gateway” is used instead.
HTTP ports – A finite list of port numbers used to access content of the WWW. Such a list typically includes port 80, 443 and possibly 8080 and similar numbers.
Local Area Network – Network which has been encapsulated from the Internet by firewalls. The LC is located within the LAN. The TOE client runs on the LC, while the TOE server runs on the TOE host, which is situated in the DMZ.
Local computer – A computer in a LAN with controlled access to the Internet. A local computer is used by one or several users for completion of their tasks.
Malware – A program (which might be an active content) which performs actions without explicit consent by the user under which environment it is launched. This term includes both remote controlled as well as autonomous programs.
Net device – All machines connected to a network which can either or both receive and transmit data, e.g. LC, routers, switches.
Protocol Data– Data generated by the TOE or TOE host intended for audit, e.g. user name, access times and URLs of requested web pages.
ReCoBS server – This term denotes the combination of the TOE host and the TOE server. It is

taken from the BSI concept [1] but not used within this PP.
TOE Client – The LC computer that will connect to the TOE Host.
TOE Host – The physical machine that will be running the TOE Server software.
TOE Server – The server side of the TightGate-Pro (CC) Version 1.4 software
TOE Protocol – The VNC protocol modified to run through TLS to communicate with the TOE Host. This term refers to the common criteria certified version of the m-privacy TightGate-Pro (CC) Version 1.4 product.

8.2 II ACRONYMS

CC	Common Criteria
BSI	Bundesamt für Sicherheit in der Informationstechnik (Federal Office for Information Security)
DMZ	Demilitarised Zone
DOS	Denial of Service
I&A	Identification and Authentication
IT	Information Technology
HTML	Hypertext Markup Language
HTTP	Hypertext Transmission Protocol
HTTPS	Hypertext Transmission Protocol Secure
LAN	Local Area Network
LC	Local Computer
OSP	Organisational Security Policy
PDF	Portable Document Format
PP	Protection Profile
ReCoBS	Remote-Controlled Browsers System
SME	Small and Medium sized Enterprise
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Function
WWW	World Wide Web