Electronic counting of paper votes software

Security Target

EAL 2

DOCUMENT HISTORY

Version	Date	Author	Comments
0.1	2010-03-12		Initial draft
0.8	2010-04-09		Updated with SFRs and TOE Summary Specification
0.8.1	2010-04-13		Updated chapter 1.2 and 1.3.
0.9	2010-04-15		Updated document after QA comments given 2010-04-15.
0.9.1	2010-04-30		Updated document after QA comments given 2010-04-22.
0.9.2	2010-11-23		Updated Security Target after changes to the System Architecture documents.
0.9.3	2010-11-29		Updated Security Target after comments given
1.0	2010-12-13		QA

QUALITY ASSURANCE

Version	Date	QA responsible	Comments
0.8.1	2010-04-15		Some definitions stated in chapter 1.2 and 1.3 should be clarified, use ITSEF in cover page, e-vote and eVote/p-vote and pVote definitions should be consequent.
0.9	2010-04-22		Comments given to functional descriptions in chapter 1.2 and 1.3 in accordance with the latest changes defined by KRD.
0.9.2	2010-11-25		Comments given to mechanisms regarding authentication and authorization

1 ST Introduction

1.1 ST and TOE Identification

1.1.1 ST Reference

ST Identification	E-counting of p-votes software Security Target
ST Version	1.0
ST Publish Date	2010-12-13

Table 1-1: ST reference

1.1.2 TOE Reference

TOE Identification	E-counting of p-votes software
TOE Developer	ErgoGroup
TOE Version	1.0
TOE Date	03.06.2011

Table 1-2: TOE reference

1.2 TOE Overview

This TOE is a software product designed to electronically tally paper votes (p-votes) as a part of a voting scheme for conducting both paper based and electronic elections. This TOE includes configuring automated counting solutions and handling these results, and the actual scanning and counting of p-votes. The results from the tallying are reported to an external election administration system. The e-voting scheme is divided into three domains; e-voting-, administration- and p-voting domain, where the p-voting domain is covered by this TOE.

An overview of how the TOE interacts with other systems in the e-voting solution is shown in Figure 1-1.

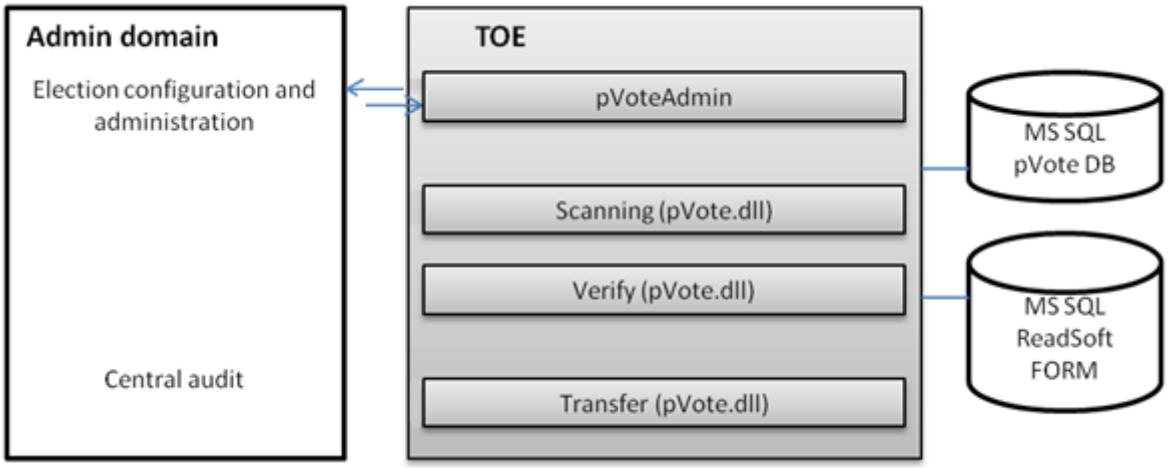

Figure 1-1: TOE interaction overview

The basic security functionalities that cover the different steps of the election included in the TOE are:

- Authorization of election officials (use of the security token provided by Election administration software)
- Configuration of e-counting system
- Scanning process
- Verify process
- Counting process
- Digitally signing of counting results
- Delivery of counting results
- TOE audit data generation

The election administration system provides the authorization mechanism to create a security token that is used by e-counting of p-votes system. This token enables the system to benefit from the role based access control provided by the Administration domain. The security token is digitally signed by the administration system's key and encrypted with a key derived from a user provided password. It's entirely up to the receivers of the security tokens to verify it and comply with the information it contains.

Authentication and authorization overview is shown in Figure 1-2.

Figure 1-2: Authentication and authorization overview

1.2.1 Overview of the p-voting process

The voter collects her ballot and makes corrections if wanted. The voter then identifies herself to the polling station committee. The committee checks the electoral roll. This may be done by a bar-code reader reading the polling card or by finding the voter in the electoral roll if the voter has not brought the polling card.

If the voter is entitled to vote, the electoral roll is updated, the paper ballot is stamped and the ballot is submitted into the ballot box.

The electoral roll will be national and available to all polling stations. This will allow the voters to present themselves at any polling station in the municipality. It must however be decided if the counting still needs to be done at the voting district level. The system supports both registration of votes and counting of votes at the same station as well as registration in one station and counting at another station.

Voters who are already marked in the electoral roll i.e. having submitted p-votes in advance (and are not entitled to vote according to the rules above) or voters, who are not found in the electoral roll, can be registered and allowed to cast their vote. However, these votes are registered and kept separate. The system supports the process to accept or reject such votes (votes in special cover). According to RPA § 10-1 (1) a. – c.

Any paper vote cast by a voter in advance or on the election day, will annul any electronic votes cast by that voter.

1.3 TOE Description

The e-counting of p-votes system facilitating the different tasks required for digitalizing paper ballots and producing text data from the images. The e-counting of p-votes software will handle the scanning, verifying and electronic counting process, and report the results to the electronic elections administration system. Election configurations are transferred to the p-voting domain from the administrative domain outside the scope of this TOE.

1.3.1 Scanning process

Before start scanning paper votes (ballots) the relevant Committee must decide what is to be scanned (e.g. selection type, location). Each location can be scanned in total or in parts. The “ReadSoft Scan”-software facilitates batch scanning. Each batch is preceded with a form indicating batch identification

and location. The form is interpreted by the software and the images are marked with the batch information as metadata in the ReadSoft database. The “batch form” can either be preprinted or manually filled out by the operator. If further Meta Data are required, a dialog will be displayed for manually entering at the beginning of each scan.

Paper based votes may be scanned at the scanning centers, and signed EML files are produced at each location. These files are uploaded by the local election officials through the administration interface. Access to the system is handled through a separate authentication process.

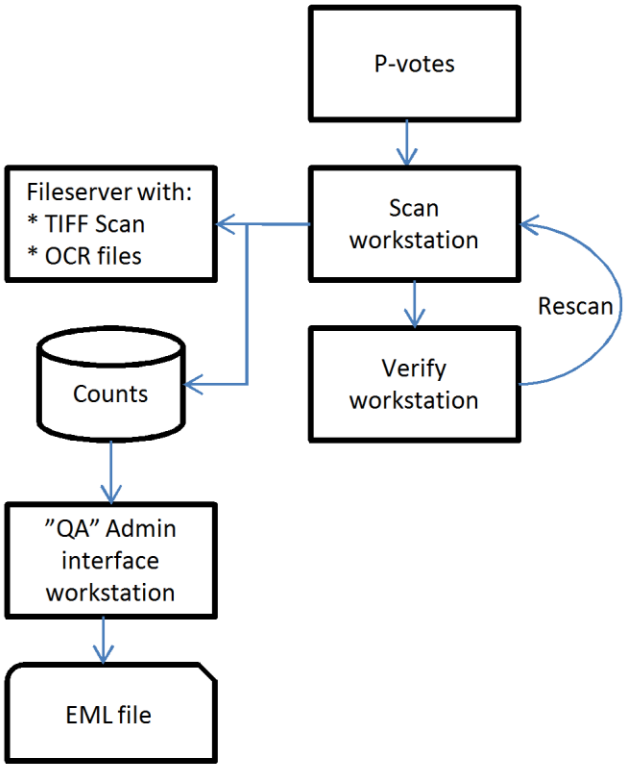

Figure 1-3: Scanning of p-votes process

The paper ballots are scanned using a document scanner with Automatic Document Feeder (ADF). The scanner produces TIFF-images.

“ReadSoft Scan” has the possibility to OCR the images real-time without loss of scanner speed. This eliminates the use of a post-scanning OCR-engine.

1.3.2 Verify process

The images are stored on a file share and the metadata and values from the ballot is stored in a central repository (DB). The system is able to recognize minimum 99% of all scanned ballots correctly. If the ballot is unrecognized, it is transferred to a special queue for manual verification on the “Verify workstation”.

The system checks that each ballot ID is unique and prompts the user if a duplicate is detected. The operator is prompted on the “Verify workstation”. The system can be set up to discard duplicates automatically.

Verification is performed by the operator by side-by-side comparison of image files and interpreted data. The operator confirms or rejects values according to the rules defined by the EC.

The system will use the “ReadSoft Verify” software to present the EC with rejected ballots. If the ballot is unrecognized by the system, the EC, or the operator appointed by the EC, can approve or reject the ballot according to the current legislation. The ballot will be shown as a high quality image of both sides of the rejected ballot. This can easily be compared to the paper copy. All rejected ballots will be logged and are saved for future reference.

It is possible for an operator to search for specific ballot IDs and display the ballot images. The ballot ID is a unique identifier located on each ballot. This can be used to verify if the ballot has been scanned previously. The purpose of the search is to find ballots that could be true duplicates, as a result of scanner operator error. If a duplicate ID is found, the ballots in question can be checked to verify if they are indeed the same. To verify identical ballots the changes, stamp and other marks on the ballot is checked.

The system is able to scan and interpret special ballots for blind, provided that party information is printed on the ballot. It is important that the ballots do not show if it has been submitted by a visually impaired person or not. This is to support the anonymity of the visually impaired person. This is particularly important on the election day as it is likely to be few visually impaired voters in each voting district.

Thus, if ballots with Braille signs are used, the ballot should always also contain the same information in regular characters (so that you can never tell if the actual ballot were submitted by a visually impaired person or not). Thus, normal OCR-scanning can always be used to scan even ballots submitted by a visually impaired person.

1.3.3 Transfer of count results to the Administrative service

The application pVoteAdmin performs the following tasks in order to provide the election administration system with the counts from the p-vote domain:

- **Generate EML files of counting results;** Counts and recounts are applied to the Counting database as shown in **Figure 1-3**. The pVoteAdmin uses these data to produce an EML file.

- **Sign EML file;** All the EML files exchanged between the domains must be signed. It’s the applications that create the EML file that is responsible for creating the signature. The signature is stored in an external file. As a result of this the EML file and signature file are always exchanged together. Communication between the modules inside and outside the boundaries of the TOE must be properly secured. The digital signing is performed by one election officials using her smart card.

- **Upload EML file through a web service on the central administration system;** The counting results are transferred to the election administration system.

1.4 TOE Boundaries

1.4.1 Physical Boundaries

TOE physical boundaries include e-counting of p-votes software, and all relevant guidance documentation.

The following table identifies software components and indicates whether or not each component is in the TOE:

TOE or Environment	Component	Description
TOE	e-counting of p-vote software v.xx	Software components that provides e-counting functionality to the p-voting domain. It is also able to handle security tokens provided by the Administrative system to maintain requirements regarding authorization.
Environment	External administrative system	Central election administrative and settlement system, including central audit system. The Administrative domain also provides security tokens used for authorization by e-counting of p-votes system
Environment	Microsoft Active Directory	Directory service providing authentication.
Environment	Operating system	e.g. Windows XP, Vista, 7, Windows server
Environment	Microsoft SQL Server 2008 Database	Database
Environment	Web browser	e.g. Internet Explorer, Mozilla Firefox
Environment	JasperReport	Report generator
Environment	Scanner (with ADF)	e.g. ISIS, Cofax scanner
Environment	ReadSoft Documents for forms version 5.2	Scanning software (ICR/OCR capable)

Table 1-3: TOE and Environment components

Authentication mechanisms are provided by standard Microsoft Active Directory outside the scope of this TOE. Access control mechanisms for restricting access to applications and services on the e-counting of p-vote system are handled by security tokens provided by the Administrative system.

All communications between TOE and modules outside the boundaries of the TOE are properly secured. Data exchanged between the domains are performed by using the Election Markup Language (EML).

1.4.2 Logical Boundaries

TOE logical boundaries include all software components inside the e-counting of p-votes software. The following Security Functions are provided by the TOE.

- Security audit
- Identification and authorization
- Cryptographic support
- User data protection

1.4.3 External TOE Components

There TOE interacts with the following external election system components:

- Election Administration and configuration system.
- Microsoft Active Directory.

1.5 Document Conventions

The notation, formatting and conventions used in this ST are consistent with version 3.1 R3 of the Common Criteria (CC).

Assumptions: TOE secure usage assumptions are given names beginning with “A.”, e.g. A.ACCESS.

Threats: Threats are given name beginning with “T.”, e.g. T.MALFUNCTION

Policies: Organizational Security Policies are given names beginning with “P.”, e.g. P.AUDIT

Objectives: Security Objectives for the TOE and the TOE environment are given names beginning with “O.” and “OE.”, respectively e.g. O.AUTHENTICATION and OE.TRUSTED ADMINISTRATOR

1.6 Document Terminology

Please refer to CC part 1 Section 4 for definitions of commonly used CC terms and Section 5 for a complete list of abbreviated terms used in CC.

1.6.1 ST Specific Terminology

Audit Service	Gathers all auditable information, ensures integrity of the auditable data and secures log information in an immutable format.
Authentication	The provision of assurance of the claimed identity of a person or data.
Authentication data	Information used to verify the claimed identity of a user.
Auditor	A user reviewing the audit data with tools in the TOE or its environment.
Ballot Box	Where the ballots are stored until being counted. The ballot box can be physical and electronic.
Ballot template	A template of a vote in which people select a candidate in an election.
Digital signature	Data appended to, or a cryptographic transformation of, a data unit that allows a recipient of the data unit to prove the source and integrity of the data unit and protect against forgery e.g. by the recipient.
Election Service Domain	Government controlled infrastructure containing; Election Preparation Domain, Voting Support Domain, E-voting Collection Domain, Electoral Roll Domain, Paper Voting Domain, Election Settlement Domain, E-vote Counting Domain and Audit Domain.
E-voting	An e-election or e-referendum that involves the use of electronic means in at least the casting of the vote.
Hardware Security Module	Cryptographic module used to generate the signature in qualified certificates and which are represented in the TOE.
Message Authentication Code	A message authentication code (MAC) is a short piece of information used to authenticate a message.
Paper Voting Domain	The Paper Voting Domain (PVD) includes the infrastructure to perform p-vote scanning and counting.
PKCS#12	Defines a file format commonly used to store private keys with accompanying public key certificates, protected with a password-based symmetric key. PFX is a predecessor to PKCS#12. This container format can contain multiple embedded objects, e.g. multiple certificates. Usually protected/encrypted with a password. Can be used as a format for the Java key store.
P-voting	If the voter is entitled to vote, the electoral roll is updated, the paper ballot (p-vote) is stamped and the ballot is submitted into the ballot box.
Signature-verification data (SVD)	Data, such as codes or public cryptographic keys, which are used for the purpose of verifying an electronic signature.
User data	Data created by and for the user that does not affect the operation of the TSF.
Zero Knowledge Proof	Verification to prevent requiring access to the private key used to encrypt the information for verifying the correctness of the information

Table 1-4: ST specific terminology

1.6.2 Acronyms

ADF	Automatic Document Feeder
CAI	Common Authentication Infrastructure
CC	Common Criteria
EAL	Evaluation Assurance Level
EML	Election Markup Language
ESD	Election Service Domain
ICR	Intelligent Character Recognition
MAC	Message Authentication Code
OCR	Optical Character Recognition
SFP	Security Function Policy
TOE	Target of Evaluation

Table 1-5: Acronyms

2 Conformance Claim

The E-counting of p-vote software Security Target has been developed using the Common Criteria (CC) Version 3.1 R3 (Evaluation Criteria for Information Technology Security; Part 1: Introduction and general model, Part 2: Security functional requirements, and Part 3: Security assurance requirements).

The TOE is Part 2 conformant, Part 3 conformant, and meets the requirements of EAL2.

This ST does not claim conformance to any PPs.

3 Security Problem Definition

3.1 Secure Usage Assumption

Assumptions	Description
A.Authentication	It is assumed that user identification and authentication shall be effective before any action of the TOE can be carried out.
A.Contigency Plan	It is assumed that there are provided a documented plan to maintain continuity of operation in an emergency or disaster.
A.Network	It is assumed that all connections to peripheral devices reside within the controlled access facilities.
A.Physical	It is assumed that TOE servers and hardware are installed in a physically secure location that can only be accessed by authorized users.
A.Scanning	It is assumed that all paper votes are scanned accurately with proper character recognition.
A.Secure Installation and Operation	It is assumed that operating system and other required software of the TOE is installed and managed in a secure way.
A.Timestamp	It is assumed that the TOE environment provide reliable synchronized time sources.
A.Trusted Administrator, Operators	It is assumed that the administrator and operators are non-hostile, well trained and following all administrator and user guidance.

Table 3-1: Secure Usage Assumption

3.2 Threats

The asset under attack is the information transiting the TOE. In general, the threat agent is most likely people with TOE access (who are expected to possess average expertise, few resources and moderate motivation) or failure of the TOE or peripherals.

The following items detail threats in an enterprise network which the TOE is intended to address:

Threats	Description
T.Counting Failure	The TOE may incorrectly count votes.
T.Malfunction	Users can cause malfunction like re-installation, and/or initialization of the e-counting software.
T.Management	Administrator can threat the TOE security by insecure management, configuration and operation
T.Modification of private/secret keys	A private/secret key is improperly disclosed or modified.
T.Non Integrity	Lack of integrity of the User or TSF data may result in altered information by unauthorized persons in a way that is not detectable by authorized users.
T.Unauthorized System Modification	Unauthorized modifications of the system, affecting operational capabilities, can occur.
T.Unauthorized voting	Duplicate or fraudulent vote can occur.
T.Unexpected Events	Data may be lost by unexpected events like hardware, software and/or storage devices fault.

Table 3-2: Threats

3.3 Organizational Security Policies (OSP)

The TOE is compliant with the applicable parts of:

The Council of Europe Recommendation Rec(2004)11
The Representation of the People Act (RPA)

The following organizational security policies are tailored from E-vote 2011 Security Objectives defined by Norwegian ministry of local government and regional development.

Organizational Policies	Description
P.Administrator	Administrator rights must only be given to authorized election officials.
P.Audit	The TOE must audit every auditable event and keep the audit record secure. Audit records are protected from unauthorized access and do not pose any security risk of the voter anonymity.
P.Authorized Users	Users must be authorized before interacting with the TOE.
P.Data Authentication	Voting data must be authenticated to verify its integrity.
P.Test	Observers must be provided an opportunity to have access to relevant software information to see physical and electronic safety measures for servers and verify that ballot box and that votes are being counted.

Table 3-3: Organizational Security Policies

4 Security Objectives

4.1 Security Objectives for the TOE

This section defines the IT Security Objectives that are to be addressed by the TOE.

Security Objective	Description
O.Audit	The TOE must provide a means to record readable audit records, with accurate dates, time and events. Furthermore, the TOE must provide variable manners to refer audit record.
O.Authorization	Administrator, operator and election observer's privileges shall be the minimum necessary to satisfy the operational requirements for the e-counting of p-votes system.
O.Counting	The TOE shall accurately count the votes and the counting shall be reproducible.
O.Data Protection	No user data will be permanently lost in the event of TOE breakdown.
O.Duplicate	The TOE must prevent duplicate counting of p-votes.
O.Integrity	Digital signatures must be used to provide integrity of data communicated between TOE and external components. Cryptographical keys must be managed in a secure way.
O.Manage	The TOE must provide manners that maintain the TOE secure to defined user roles of the election system.
O.Self Protection	The TOE must protect itself against attempts by unauthorized users to bypass, deactivate, or tamper with security functions.
O.Test	The TOE must support testing of its security functions.
O.TSF Data Protection	The TOE must protect TSF data from unauthorized exposure, alteration and deletion.
O.Verify	It shall be possible to certify that the TOE act accordingly to its specification.

Table 4-1: Security Objectives for the TOE

4.2 Security Objectives for the Operational Environment

This section defines the IT Security Objectives that are to be addressed by the Operational Environment of the TOE.

Security Objective for Environment	Description
OE.Contigency Plan	There shall be provided a documented plan to maintain continuity of operation in an emergency or disaster.
OE.Physical	Appropriate physical security must be provided for the TOE
OE.Identification and Authentication	User identification and authentication shall be effective before any action of the TOE can be carried out.
OE.Install	The TOE is delivered, installed, managed and operated in a manner that maintains the security objectives.
OE.Management	The TOE must be managed in a way that maintains security policies.
OE.Modify	Only properly authorized users shall have the possibility to delete or modify any votes handled by the TOE.
OE.Non Disclosure	Disclosure of audit information to unauthorized persons shall be prevented.

OE.Timestamp	The TOE environment must provide reliable synchronized time sources.
OE.Trusted Administrator, Operators	Authorized administrator and operators must be trained as to establish and maintain security policies in practice.

Table 4-2: Security Objectives for the Operational Environment

4.3 Security Objective Rationale for the TOE

4.3.1 Mapping of Security Objectives to threats and organizational security policies

The following table represents a mapping of the Threats and Organizational Security Policies to the Security Objectives defined in this ST. All Security Objectives for the Operational Environment are considered to be Secure Usage Assumptions.

	O.Audit	O.Authorization	O.Counting	O.Data Protection	O.Duplicate	O.Integrity	O.Manage	O.Self Protection	O.Test	O.TSF Data Protection	O.Verify
T.Counting Failure			X		X						
T.Malfunction		X						X			
T.Management		X					X				
T.Modification of private/secret keys										X	
T.Non Integrity						X		X			
T.Unauthorized System Modification	X	X						X		X	
T.Unauthorized voting					X						
T.Unexpected Events				X							
P.Administrator		X									
P.Audit	X										X
P.Authorized Users		X									
P.Data Authentication						X					
P.Test									X		

Table 4-3: Mapping of Security Objectives to threats and organizational security policies

4.3.2 Justification of Security Objectives to threats and OSPs

Each threat and OSP included in this ST is covered by the following Security Objectives as explained below.

O.Audit	Threats/OSPs countered: **T.Unauthorized System Modification** and **P.Audit**
	The security objective **O.Audit** covers **P.Audit** in a manner that the TOE audits every auditable event and keeps the audit record secure. The audit record is protected from unauthorized access and do not pose any security risk of the voter anonymity. **T.Unauthorized System Modification** is covered by the TOE generating audit data which are sent to a central audit repository for real time analysis and alarm generation.

O.Authorization	Threats/OSPs countered: **T.Malfunction**, **T.Management**, **T.Unauthorized System Modification**, **P.Administrator** and **P.Authorized Users**
	The security objective **O.Authorization** cover the threats **T.Malfunction, T.Management, T.Unathorized System Modification, P.Administrator** and **P.Audit** in a manner that authorization framework will be used for management of securable objects. The selected framework will provide the ability to integrate with the Administrative system of eVote for role based access to securable objects.

O.Counting	Threats/OSPs countered: **T.Counting Failure**
	The security objective **O.Counting** covers the threat **T.Couting Failure** in a manner that the TOE provides verifiable counting results, which also is reproducible. The quality assured counting results are digitally signed before being sent to the admin domain.

O.Data Protection	Threats/OSPs countered: **T.Unexpected Events**
	The security objective **O.Data Protection** covers **T.Unexpected Events** in a matter that no user data will be permanently lost in the event of the TOE breakdown. Data transactions are designed to support rollback in case of failure to keep the information in a consistent status.

O.Duplicate	Threats/OSPs countered: **T.Counting Failure** and **T.Unauthorized voting**
	The security objective **O.Duplicate** covers the threat **T.Counting Failure** and **T.Unauthorized voting** by preventing duplicate counting of p-votes is the threat of counting errors and unauthorized voting diminished. The p-votes cast on the election day are subject to manual routines for checking the voters' eligibility and preventing duplicate ballots being cast in the ballot box. The scanning software (Readsoft) checks the ballotID and automatically removes duplicate ballots.

O.Integrity	Threats/OSPs countered: **T.Non Integrity** and **P.Data Authentication**
	The security objective **O.Integrity** covers **T.Non Integrity** and **P.Data Authentication** because; verified counting data is digitally signed to ensure that the integrity of the data is intact when sent to other modules in the election system.

O.Manage	Threats/OSPs countered: **T.Management**
	The security objective **O.Manage** covers the threat **T.Management** by providing the management roles (i.e. scanning officer) with the least amount of privileges necessary to perform the tasks intended. **T.Management** is mitigated with authorization and auditing.

O.Self Protection	Threats/OSPs countered: **T.Malfunction**, **T.Non Integrity** and **T.Unauthorized System Modification**
	The security objective **O.Self Protection** cover the threats **T.Malfunction**, **T.Non Integrity** and **T.Unauthorized System Modification** by only allowing authorized personnel access the TOE and its environment.

O.Test	Threats/OSPs countered: **P.Test**
	The security objective **O.Test** covers the policy **P.Test** because the TOE and its associated documentation will demonstrate that it is in an accurate implementation.

O.TSF Data Protection	Threats/OSPs countered: **T.Modification of private/secret keys** and **T.Unauthorized System Modification**
	The security objective **O.TSF Data Protection** covers the threats **T.Modification of private/secret keys** and **T.Unauthorized System Modification** since it states that it must protect TSF data from unauthorized exposure, alteration and deletion.

O.Verify	Threats/OSPs countered: **P.Audit**
	The security objective **O.Verify** covers the policy **P.Audit** by making it possible to verify that the TOE act accordingly to its specification by analyzing audit data.

4.4 Security Objective Rationale for the Operational Environment

4.4.1 Mapping of Operational Environment Security Objectives to assumptions

	OE.Contigency Plan	OE.Physical	OE.Identification and Authentication	OE.Install	OE.Management	OE.Modify	OE.Non Disclosure	OE.Timestamp	OE.Trusted Administrator, Operators
A.Authentication			X			X	X		
A.Contigency Plan	X								
A.Network		X					X		
A.Physical		X				X			
A.Scanning					X				
A.Secure Installation and Operation				X	X	X			X
A.Timestamp								X	
A.Trusted Administrator, Operators			X			X			X

Table 4-4: Mapping of Operational Environment Security Objectives to assumptions

4.4.2 Justification of Operational Environment Security Objectives and assumptions

All of the Security Objectives for the Operational Environment are considered to be Secure Usage Assumptions.

OE.Contigency Plan	Assumption countered: **A.Contigency Plan**
	A.Contigency Plan assume that there are provided a documented plan to maintain continuity of operation in an emergency or disaster. This is covered by the objective **OE.Contigency Plan** which states that there shall provided a documented plan to maintain continuity of operation in an emergency or disaster.

OE.Physical	Assumption countered: **A.Network** and **A.Physical**
	A.Physical and **A.Network** assumed that TOE servers are installed in a physically secure location that can only be accessed by authorized users. Further it is assumed that all connections to peripheral devices reside within the controlled access facilities. The objective **OE.Physical** ensures that appropriate physical security must be provided for the TOE.

OE.Identification and Authentication	Assumption countered: **A.Authentication** and **A.Trusted Administrator, Operators**
	OE.Identification and Authentication covers the assumptions **A.Authentication** and **A.Trusted Administrator, Operators** because it states that identification and authentication shall be effective before any action of the TOE can be carried out.

OE.Install	Assumption countered: **A.Secure Installation and Operation**
	A.Secure Installation and Operation is covered by the objective **OE.Install** because it requires the TOE to be delivered, installed in a manner that maintains the security objectives.

OE.Management	Assumption countered: **A.Scanning** and **A.Secure Installation and Operation**
	A.Secure Installation and Operation and **A.Scanning** is covered by the objective **OE.Management** because it states that the TOE shall be managed in a manner that maintains the security objectives and that the scanning process is performed accurately with proper character recognition.

OE.Modify	Assumption countered: **A.Physical**, **A.Authentication**, **A.Secure Installation and Operation** and **A.Trusted Administrator, Operators**
	A.Physical and **A.Authentication** are assumptions about access restrictions to TOE and its environment. These and assumptions regarding secure management (**A.Secure Installation and Operation** and **A.Trusted Administrator, Operators**) are covered by **OE.Modify** which states that only properly authorized users shall have the possibility to delete or modify any votes handled by the TOE.

OE.Non Disclosure	Assumption countered: **A.Authentication** and **A.Network**
	Disclosure of audit information to unauthorized persons shall be prevented (**OE.Non Disclosure**). This objective covers the assumption **A.Authentication,** since no actions shall be allowed for users before they are properly authenticated. Further, this objective covers **A.Network** since it states that all connections to peripheral devices reside within the controlled access facilities.

OE.Timestamp	Assumption countered: **A.Timestamp**
	OE.Timestamp covers the assumption **A.Timestamp** because it states that the TOE environment shall provide reliable synchronized time sources for TOE.

OE.Trusted Administrator, Operators	Assumption countered: **A.Secure Installation and Operation** and **A.Trusted Administrator, Operators**
	OE.Trusted Administrator, Operators states authorized administrator and operators must be trained as to establish and maintain security policies in practice. This cover the assumptions about the administrator and operators to be non-hostile, well trained and follow all administrator and user guidance (**A.Trusted Administrator, Operators**). And secure installation (**A.Secure Installation and Operation**).

5 Security Requirements

5.1 TOE Security Functional Requirements

The SFRs defined in this section are taken from part 2 of the CC.

5.1.1 Security Audit (FAU)

5.1.1.1 Audit data generation (FAU_GEN.1)

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:
a) Start-up and shutdown of the audit functions;
b) All auditable events for the [*basic*] level of audit; and
c) [*Unsuccessful attempts to read information from the audit records, Election transactions, Attacks on the e-counting of p-voting system, Modifications in the behaviour of TSF functions, Modifications to the group of users that are part of a role, System failure and malfunctions*].

FAU_GEN.1.2 The TSF shall record within each audit record at least the following information:
a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and
b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [*location of where the event was generated*].

5.1.1.2 User identity association (FAU_GEN.2)

FAU_GEN.2.1 For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

5.1.1.3 Audit review (FAU_SAR.1)

FAU_SAR.1.1 The TSF shall provide [*System administrator, Auditor and Observers*] with the capability to read [*Generated audit data (FAU_GEN.1)*] from the audit records.

FAU_SAR.1.2 The TSF shall provide the audit records in a manner suitable for the user to interpret the information.

5.1.1.4 Restricted audit review (FAU_SAR.2)

FAU_SAR.2.1 The TSF shall prohibit all users read access to the audit records, except those users that have been granted explicit read access.

5.1.2 Communication (FCO)

5.1.2.1 Enforced proof of origin (FCO_NRO.2)

FCO_NRO.2.1 The TSF shall enforce the generation of evidence of origin for transmitted [*audit records, election transactions, TSF functions and users that are part of a role*] at all times.
FCO_NRO.2.2 The TSF shall be able to relate the [*Digital signature*] of the originator of the information, and the [*Message Authentication Code (MAC)*] of the information to which the evidence applies.
FCO_NRO.2.3 The TSF shall provide a capability to verify the evidence of origin of information to [*originator*, [*election admin system*]] given [*the originators public key*].

5.1.3 Cryptographic support (FCS)

5.1.3.1 Cryptographic key destruction (FCS_CKM.4)

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [*overwrite*] that meets the following: [*none*].

5.1.3.2 Cryptographic operation (FCS_COP.1)

FCS_COP.1.1 The TSF shall perform [*decryption of "security token" imported from the admin domain*] in accordance with a specified cryptographic algorithm [*AES*] and cryptographic key sizes [*256 bits*] that meet the following: [*FIPS 197*].

5.1.4 User Data Protection (FDP)

5.1.4.1 Complete Access Control (FDP_ACC.2)

FDP_ACC.2.1 The TSF shall enforce the [*Role Based Access Control (RBAC)*] on [*all program components, system configuration files, system keys, election configurations, logs, electronic ballot boxes and counting results*] and all operations among subjects and objects covered by the SFP.

FDP_ACC.2.2 The TSF shall ensure that all operations between any subject controlled by the TSF and any object controlled by the TSF are covered by an access control SFP.

5.1.4.2 Security Attribute based access control (FDP_ACF.1)

FDP_ACF.1.1 The TSF shall enforce the [*Role Based Access Control (RBAC)*] to objects based on the following: [*user identity and user access permissions*].
FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [*if the user has been explicitly granted access to the objects*].
FDP_ACF.1.3 The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [*no rules*].

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the following additional rules: [*Access by a user to applications that are not permitted shall be denied*].

5.1.4.3 Data Authentication with Identity of Guarantor (FDP_DAU.2)

FDP_DAU.2.1 The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of [*audit records, election transactions, TSF functions and users that are part of a role*].

FDP_DAU.2.2 The TSF shall provide [*System administrator, Operator and Observers*] with the ability to verify evidence of the validity of the indicated information and the identity of the user that generated the evidence.

5.1.4.4 Basic rollback (FDP_ROL.1)

FDP_ROL.1.1 The TSF shall enforce [*Role Based Access Control (RBAC)*] to permit the rollback of the [*information transactions in case of component- or network failure*] on the [User data and TSF data].

FDP_ROL.1.2 The TSF shall permit operations to be rolled back within the [*scope of current transaction, temporary disconnections shall be supported*].

Application Note: The RBAC system provides the ability to restrict permission to performing rollback operations to well defined roles.

5.1.5 Identification and authentication (FIA)

5.1.5.1 User attribute definition (FIA_ATD.1)

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users: [*user identifier and roles*].

5.1.5.2 User authentication before any action (FIA_UAU.2)

FIA_UAU.2.1 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

5.1.5.3 User identification before any action (FIA_UID.2)

FIA_UID.2.1 The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

5.1.6 Protection of the TSF (FPT)

5.1.6.1 Inter-TSF detection of modification (FPT_ITI.1)

FPT_ITI.1.1 The TSF shall provide the capability to detect modification of all TSF data during transmission between the TSF and another trusted IT product within the following metric: [*modifications detected by a standard cryptographic hash function and digital signature*].

Application Note: Although the TOE performs an integrity verification function, the hashing algorithm used in the verification is not directly implemented in the TOE. The TOE makes use of the environmental cryptographic libraries to perform this function.

FPT_ITI.1.2 The TSF shall provide the capability to verify the integrity of all TSF data transmitted between the TSF and another trusted IT product and perform [*discarding of TSF data*] if modifications are detected.

5.1.6.2 Reliable time stamp (FPT_STM.1)

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps.

5.2 Security Functional Requirements dependencies

Requirement	Dependencies	Dependency met
FAU_GEN.1	FPT_STM.1	Yes
FAU_GEN.2	FAU_GEN.1 and FIA_UID.1	Yes, FIA_UID.2 is hierarchical to FIA_UID.1
FAU_SAR.1	FAU_GEN.1	Yes
FAU_SAR.2	FAU_SAR.1	Yes
FCO_NRO.2	FIA_UID.1	Yes, FIA_UID.2 is hierarchical to FIA_UID.1
FCS_CKM.4	FCS_CKM.1	No, key data generation are provided by external CA service.
FCS_COP.1	FCS_CKM.4	Yes
FDP_ACC.2	FDP_ACF.1	Yes
FDP_ACF.1	FDP_ACC.1 and FMT_MSA.3	Yes, FDP_ACC.2 is hierarchical to FDP_ACC.1. No, FMT_MSA.3 is not required since management functions are provided by the administration system outside the scope of this TOE.
FDP_DAU.2	FIA_UID.1	Yes, FIA_UID.2 is hierarchical to FIA_UID.1
FDP_ROL.1	FDP_ACC.1	Yes, FDP_ACC.2 is hierarchical to FDP_ACC.1
FIA_ATD.1	None	Yes
FIA_UAU.2	FIA_UID.1	Yes, FIA_UID.2 is hierarchical to FIA_UID.1
FIA_UID.2	None	Yes
FPT_ITI.1	None	Yes
FPT_STM.1	None	Yes

Table 5-1: Security Functional Requirements dependencies

5.3 Rationale for TOE Security Requirements

5.3.1 Mapping of TOE Security Requirements to Objectives

	O.Audit	O.Authorization	O.Counting	O.Data Protection	O.Duplicate	O.Integrity	O.Manage	O.Self Protection	O.Test	O.TSF Data Protection	O.Verify
FAU_GEN.1	X				X						X
FAU_GEN.2	X							X			
FAU_SAR.1	X	X									
FAU_SAR.2		X									
Feil! Fant ikke referansekilden.						X					
FCS_CKM.4						X					
FCS_COP.1			X			X					
FDP_ACC.2		X	X	X			X				
FDP_ACF.1		X		X			X			X	
FDP_DAU.2						X					X
FDP_ROL.1				X							
FIA_ATD.1		X									
FIA_UAU.2								X			
FIA_UID.2								X			
FPT_ITI.1						X				X	X
FPT_STM.1	X										

Table 5-2: Mapping of TOE Security Requirements to Objectives.

5.3.2 Justification of TOE Security Requirements to Objectives

The following table shows how each TOE Security Requirements satisfy the Objectives defined in this ST.

FAU_GEN.1	Objectives addressed: **O.Audit**, **O.Duplicate** and **O.Verify**
	FAU_GEN.1 requires TSF to generate audit records. This requirement cover the objective for audit as defined in **O.Audit** and contribute with possibilities for inspections as described in **O.Verify**. Further, modifications in the behaviour of TSF functions will be detected by this requirement which prohibit errors that may cause failure in counting of votes (**O.Duplicate**).

FAU_GEN.2	Objectives addressed: **O.Audit** and **O.Self Protection**
	FAU_GEN.2 requires audit events (**O.Audit**) resulting from actions of identified users, to be associated with the identity of the user that caused the event. This is important to provide self protection in a matter of performing corrective actions (**O.Self Protection**).

Feil! Fant ikke referansekilden.	Objectives addressed: **Feil! Fant ikke referansekilden.** and **Feil! Fant ikke referansekilden.**
	Feil! Fant ikke referansekilden. Requires TSF to provide generated audit data to authorized user roles only which covers **Feil! Fant ikke referansekilden.**. Further TSF shall provide the audit records in a manner suitable for the user to interpret the information which covers **Feil! Fant ikke referansekilden.**.

FAU_SAR.2	Objectives addressed: **O.Authorization**
	FAU_SAR.2 TSF shall prohibit all users read access to the audit records, except those users that have been granted explicit read-access which cover **O.Authorization**.

FCO_NRO.2	Objectives addressed: **O.Integrity**
	The TOE is required to provide capability to verify the evidence of origin of information as described in **FCO_NRO.2**, by use of digital signatures for the originator of the information. This covers the objective **O.Integrity**.

FCS_CKM.4	Objectives addressed: **O.Integrity**
	FCS_CKM.4 requires TSF to destroy cryptographic keys in order to provide secure management related to the services described in **O.Integrity**.

FCS_COP.1	Objectives addressed: **O.Counting** and **O.Integrity**
	FCS_COP.1 requires TSF to perform digital signature to verify the integrity of counted votes when information are transferred between systems. This will ensure that the counted results are not modified and therefore are correct (**O.Counting**). The cryptographic algorithm as described in **FCS_COP.1** will satisfy the objective defined by **O.Integrity**.

FDP_ACC.2	Objectives addressed: **O.Authorization**, **O.Counting**, **O.Data Protection** and **O.Manage**
	FDP_ACC.2 requires TSF to ensure that all operations between any subject controlled by the TSF and any object controlled by the TSF are attached to an access control SFP; this will cover the objective **O.Authorization** and **O.Data Protection**. Further will access control SFPs as defined in this requirement ensure secure management (**O.Manage**) which decreases the risk of counting failures (**O.Counting**).

FDP_ACF.1	Objectives addressed: **O.Authorization**, **O.Manage**, **O.Data Protection** and **O.TSF Data Protection**
	FDP_ACF.1 requires the TSF to enforce functionality to determine if an operation among controlled subjects and controlled objects is allowed. The TOE shall provide authorization mechanisms to support minimal access privileges for each of the defined user roles which cover the objective **O.Authorization**. Access control mechanisms as defined will ensure protection of both user data and TSF data and therefore cover the objectives **O.Data Protection** and **O.TSF Data Protection**. Further, access control SFPs as defined in this requirement will ensure secure management (**O.Manage**).

FDP_DAU.2	Objectives addressed: **O.Integrity** and **O.Verify**
	FDP_DAU.2 requires that TSF shall provide a capability to guarantee the validity of data handled by the system; this includes all election transactions, attacks on the operation of the election system and its communications infrastructure. This requirement cover the objective **O.Integrity** since digital signatures are used to provide integrity of data communicated between TOE and external components. Further it is required that the TOE can verify evidence of the validity of the indicated information and the identity of the user that generated the evidence, which covers the objective **O.Verify**.

FDP_ROL.1	Objectives addressed: **O.Data Protection**
	FDP_ROL.1 requires the TOE to be able to retransmit user data- and TSF data transactions in case of component- or network failure. This requirement is supporting the objective **O.Data Protection** since no user data will be permanently lost in the event of TOE breakdown.

FIA_ATD.1	Objectives addressed: **O.Authorization**
	FIA_ATD.1 requires the TSF to maintain different roles belonging to individual users in order to provide required authorization mechanisms to cover the objective **O.Authorization**.

FIA_UAU.2	Objectives addressed: **O.Self Protection**
	FIA_UAU.2 requires each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user. This is required as a function for the TOE to protect itself against attempts by unauthorized users to bypass, deactivate, or tamper with security functions as stated by the objective **O.Self Protection**.

FIA_UID.2	Objectives addressed: **O.Self Protection**
	FIA_UID.2 requires each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user. This is required as a function for the TOE to protect itself against attempts by unauthorized users to bypass, deactivate, or tamper with security functions as stated by the objective **O.Self Protection**.

FPT_ITI.1	Objectives addressed: **O.Integrity**, **O.TSF Data Protection** and **O.Verify**
	FPT_ITI.1 requires that the TSF shall provide the capability to detect modification of all TSF data (**O.Verify** and **O.Integrity**) during transmission between the TSF and another trusted IT product. Modifications detected by a standard cryptographic hash function and digital signature will decrease the risk of unauthorized changes to the TOE and therefore covers the objective **O.TSF Data Protection**.

FPT_STM.1	Objectives addressed: **O.Audit**
	FPT_STM.1 requires TSF to provide reliable time stamps. This requirement cover the objective **O.Audit** where it is stated that TOE must provide a means to record readable audit records, with accurate dates and time.

5.5 Security Assurance Requirements

The Assurance Security Requirements for this Security Target are taken from Part 3 of the CC. These assurance requirements compose an Evaluation Assurance Level 2 as defined by the CC.

Assurance Class	Assurance Component	
ADV: Development	ADV_ARC.1	Security architecture description
	ADV_FSP.2	Security-enforcing functional specification
	ADV_TDS.1	Basic design
AGD: Guidance documents	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative procedures
ALC: Life-cycle support	ALC_CMC.2	Use of a CM system
	ALC_CMS.2	Parts of the TOE CM coverage
	ALC_DEL.1	Delivery procedures
ASE: Security Target evaluation	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: Tests	ATE_COV.1	Evidence of coverage
	ATE_FUN.1	Functional testing
	ATE_IND.2	Independent testing - sample
AVA: Vulnerability assessment	AVA_VAN.2	Vulnerability analysis

Table 5-3: Security Assurance Requirements for EAL2

5.5.1 Rationale for Security Assurance Requirements

EAL2 is chosen to provide a moderate level of assured security. This assurance level is consistent with the threat environment.

The assurance security requirements for this Security Target are taken from part 3 of CC.

6 TOE Summary Specification

6.1 TOE Security Functions Rationale

The TOE consists of 4 Security Functions:

- Security audit
- Cryptographic support
- Identification and authorization
- User data protection

Table 6-1 below demonstrates the tracing of TOE security functions back to aspects of the security functional requirements (SFRs).

	Security Audit	Cryptographic support	Identification and authorization	User data protection
FAU_GEN.1	X			
FAU_GEN.2	X			
Feil! Fant ikke referansekilden.	X			
FAU_SAR.2	X			
FCO_NRO.2	X			
FCS_CKM.4		X		
FCS_COP.1		X		
FDP_ACC.2			X	
FDP_ACF.1			X	
FDP_DAU.2			X	
FDP_ROL.1			X	
FIA_ATD.1			X	
FIA_UAU.2			X	X
FIA_UID.2				X
FPT_ITI.1				X
FPT_STM.1	X			

Table 6-2: Security functions to SFR mapping

The following rationale explains how the TOE is proving its Security functions.

6.1.1 Security audit

The e-counting of p-votes system components logs all significant events, recording among others user, time and event details. This includes logs of all events at all levels, attacks on the operation of the

counting system, system failures, malfunctions and other threats to the system and events at operating system level.

Log messages recorded are status/informational messages (i.e., executed transactions and their result) as well as errors/issues. All log entries contain the following information:

- Timestamp: using the clock of the computer hosting the log originator.
- Origin: the service originating the log.
- User information.
- Type: error, status, information.
- Message/event details.

6.1.2 Cryptographic support

E-counting software interacts with Bouncy Castle in order to manage secret keys in a secure way, meaning that such secret keys are always protected against unauthorized disclosure and modification using methods and standards as described in SFRs for FCS Cryptographic support.

The e-counting software is configured to automatically check the digital signature of the e-voting and p-voting data before executing them. The server does not show any screen asking for manual conformation of the execution if the signature is correct and issued by a trusted digital certificate, otherwise an error is reported and the execution aborted.

6.1.3 Identification and authorization

Identification and authorization means are based on the role based mechanism. This security control mechanism is used to prevent any non-properly authenticated and authorized user from accessing objects as described in the Security Functional Requirement (SFR) FDP_ACC.2.

The TOE maintains the roles as described in the SFR FMT_SMR.1.

Each role has some general attributes that are defined when creating the role:

- The role name and unique ID
- The owner(s) of the role
- The authentication method (i.e. the minimum authentication level required) for members of the role

6.1.4 User data protection

The e-counting software will handle election configuration provided by the administration system. This includes management of users and roles (Authorization), security audit generation and management, encryption and keys for signing, counting configuration and administration.

All attributes that describe the counting will be determined through configuration process. The Election administrator will be guided through the process, so that the configuration contains all the elements it needs. All previous counting configuration templates will be available for editing or as a basis for creating a new configuration. Manual configurations can be applied without use of previous configuration templates as well.

The Election configuration service provides export of election configuration in XML/EML format. It is possible to import either a full or partial configuration. For instance, an XML containing only parties can be imported into a new or existing configuration.