Election Administration software

Security Target

EAL 2

DOCUMENT HISTORY

Version	Date	Author	Comments
0.1	2010-04-06		Created document with adjustments for Election administration software.
0.8	2010-04-09		Added chapter 6; TOE Summary Specification
0.8.1	2010-04-13		Changed FAU_GEN.1 in chapter 5.4.
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	2010-11-15		Updated Security Target after changes to the System Architecture documents.
0.9.3	2010-11-29		Updated Security Target after comments given
1.0	2010-12-07		Added digital signature objective and 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 p-vote 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	Election administration software Security Target
ST Version	1.0
ST Publish Date	2010-12-07

1.1.2 TOE Reference

TOE Identification	Election administration software
TOE Developer	ErgoGroup
TOE Version	1.0
TOE Date	03.06.2011

1.2 TOE Overview

This TOE comprises the software product components (Admin Front End and Admin Back End) designed to provide an administration platform to conduct elections. This software must be available to election officials and administrators throughout the entire election life cycle.

Key generation, key distribution and general key management functions will be handled by a separate entity developed by Scytl. TOE communicates with an external authentication module that will in practice perform the authentication (e.g., government CAI) for TOE and its environment.

Ballot counts from the e-voting and p-voting domain will be transmitted to TOE for accumulation. The accumulated ballot counting will be handled by the settlement software of TOE.

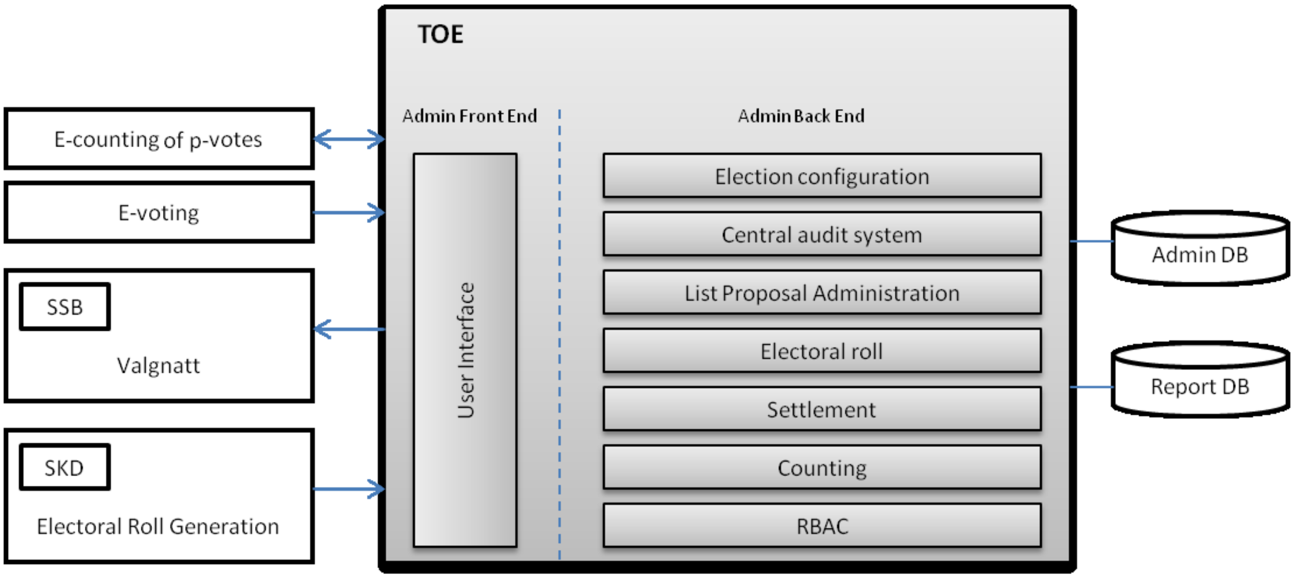

Figure 1-1: TOE and interfaces to its environment

This Administration software interacts with the e-vote and p-vote module of the election system outside the TOE. All communications between TOE and modules outside the boundaries of the TOE are properly secured. Election configuration and counting data exchanged between the domains are performed by using the Election Markup Language (EML).

The TOE is also integrated towards the Norwegian Revenue Service (SKD) such that daily updates to the Electoral Roll can be received. In addition there is an interface to send statistics (counts) to the Central Bureau of Statistics (SSB).

1.3 TOE Description

The election administrative system has functionality for defining election details such as election hierarchy, geographical hierarchy, parties and candidates, election dates and performing settlement to calculate mandates and report results to external sources.

Election configuration

The Election configuration service provides export of election configuration in XML/EML format. It is possible to import the configuration for only one selected election event or to manage several elections in parallel. The EML-files are made available for download by authorized users at the e-voting and p-voting sites.

All the EML files exchanged between the domains must be signed. The signature is stored in a separate file. As a result of this, the EML file and signature file are always exchanged together. Communications between the modules inside and outside the boundaries of the TOE must be properly secured

When the election configuration is done, the configuration is frozen after an approval procedure. At this point the configuration is versioned and extracted from the configuration database and into EML-files.

The TOE provide authorization mechanisms to ensure that a user/administrator within the ESD, once identified and authenticated, can access only those part of the system and assets necessary to perform the authorized task. Admin will generate a key to encrypt the “security token” used in the p-voting domain. This key will be generated based on a user provided password.

Audit generation

Audit data are generated from application events and stored locally in an immutabilized format. The applications log to an RSYSLOG server. The RSYSLOG server then forwards logs to the central log system. In addition, infrastructure events (OS, DB, and network) are monitored in the data center, and imported to the central log system. The central log system is high-availability. If the connection to a system or infrastructure component is broken, a local copy of the log is always available.

The logging system has interfaces towards all e-voting 2011 system components. The communication is done through the RSYSLOG protocol.

The audit system allows the electoral authorities to configure which incidents will raise an alert, allowing them to actively report and distinguish critical incidents over the regular noise on any system.

Electoral roll

Electoral roll is implemented as a database built from public data regarding eligible votes. The administration interface is implemented as part of the central election administrative system.

The system allows election officials to build and maintain an electoral roll for an election event. It provides functionality for adding or revoking access to vote, checking a voter’s eligibility, address, and other voting restrictions. It also provides interfaces for other part of the election system to check voter’s eligibility and to mark off that a voter has cast a vote in the election.

The initial data feed for the electoral roll will be received from SKD. SKD will provide daily updates on the initial data after the initial official electoral roll is created. Election officials will administer electoral roll applications.

On the election day, the officials will use the application to mark off voters as they cast their votes in the election.

The electoral roll will be exported to the e-voting system several times and will also be accessed in real-time during e-voting by means of a web-service created for this purpose.

List Proposal Administration

List proposal is a part of the central administrative system. This module consists of a public and administrative interface. The public module is a web application available to parties or organizations who wishes to be represented in an election. The applicant can submit a list of candidates and signatures that endorses the party or organization according to laws and regulations. The election officials use an administrative interface to manage the list proposals process and make the final approval of the lists.

After the approval process is finished, the list proposals convert to election configuration for parties and candidates and are transferred to other system as part of the EML-configuration.

The final approved lists of candidates will be digitally signed by the Admin system and made available for download by the e-voting and e-counting of p-votes systems.

Settlement

The settlement component provided by this TOE receives the approved election counting results of e-votes and p-votes.

This service performs the merging of e-votes and p-votes, distribution of mandates and seats according the rules imported from the setup EML. The result is a signed EML file containing the final results that is to be published.

Counting

Counting service will be provided with the results from both preliminary and final counts (outside this TOE). After the integrity and the authenticity of the counting results are verified, the approval process will start.

If the counts are equal, the Relevant Committee can approve the results. If the final count matches the preliminary count, the EC approves the count by clicking the “Approve”-button. If the counts differ, the Relevant Committee can decide to rescan the votes.

If the counts differ, the difference is shown. The information can be viewed as differ in totals, or per party by double-clicking the count.

The Operator of the Approval workstation can choose which version of the counts to compare. It is possible to delete a count by right clicking the count and choosing the option delete. It is also possible to overwrite manually. These changes will be logged in the system. The counting module of Admin also provides functionality to register manual counts of ballots.

Role Based Access Control (RBAC)

The TOE provide authorization mechanisms to ensure that a user/administrator within the ESD, once identified and authenticated, can access only those part of the system and assets necessary to perform the authorized task.

1.4 TOE Boundaries

1.4.1 Physical Boundaries

TOE physical boundaries include the servers running the Election Administration software, with CentOS operating system and patch policy sufficient for stopping all known public available vulnerabilities.

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

TOE or Environment	Component	Description
TOE	Administration software of e-voting v.xx	Software component for administrating the e-voting system
TOE	Settlement software v.xx	Settlement tool for merging of e- and p-votes and distribution of mandates
Environment	CentOS 5.5	Operating system
Environment	Glassfish Open Source Edition 3.0.1	Application server
Environment	External authentication module	The system will authenticate users in the CAI (MinID from DIFI) using SAML 2.0 over SOAP and HTTP
Environment	External key management service	Generation of the keys and certificates necessary for secure communication and to ensure integrity during exchange of data
Environment	Database PostgreSQL 9.0	Database
Environment	JasperReport	Report generator
Environment	iReport	Design and implement reports
Environment	RSYSLOG	Storage and forwarding of log messages
Environment	TPM and Tripwire	Verify integrity of system installations
Environment	Splunk	Infrastructure monitoring and alerts

Table 1-1: TOE and Environment software components

Authentication mechanisms are provided by the CAI (MinID from DIFI) outside the TOE using SAML 2.0 over SOAP and HTTP. This service is providing User accesses to the Election System application. If the user has no valid session, he/she will be transferred to the CAI using a HTTP redirect. The CAI displays a logon form and authenticates the user.

1.4.2 Logical Boundaries

TOE logical boundaries include the Admin software component. The following major Security Functions are provided by the TOE:

- Security audit
- Cryptographic support

- Identification and authorization
- Security management (Election configuration and management)

1.5 Document Conventions

The notation, formatting and conventions used in this ST are consistent with version 3.1R3 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

Auditing	It refers to the “audit stories” or the functionalities (or reports) than an election auditor would request to the application. They could be executed during or after the election.
Auditor	A user reviewing the audit data with tools in the TOE or its environment.
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.
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.
Electoral Roll	Contains the application used to verify voter eligibility and mark-off votes in the Electoral Roll.
Electoral Roll Management	Contains the IT service that is used to manage the Electoral Roll within the administrative domain. Access to read and update the Electoral Roll is handled within the administrative 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.
Logging	Includes all aspects regarding the security events registration, both the application events as the infrastructure events, and its protection through immutable logs mechanism.
Message Authentication Code	A message authentication code (MAC) is a short piece of information used to authenticate a message.
Monitoring	Refers to the tools for the monitoring activity, which have to be performed 24x7, alerting when an event (or a group of events with a specific threat pattern) has to be notified as soon as possible.
Signature-creation data (SCD)	Unique data, such as codes or private cryptographic keys, which are used by the signatory to create an electronic signature.
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.

Table 1-2: ST specific terminology

1.6.2 Acronyms

AUD	Audit Domain
CC	Common Criteria
EAL	Evaluation Assurance Level
ERD	Electoral Roll Domain
ESD	Election Service Domain
HSM	Hardware Security Module
MAC	Message Authentication Code
PBA	Poll Book Application
PP	Protection Profile
SAR	Security Assurance Requirement
SCD	Signature Creation Data
SFP	Security Function Policy
SFR	Security Functional Requirement
ST	Security Target
SVD	Signature Verification Data
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Functions
TSFI	TSF Interface
TSP	TOE Security Policy

Table 1-3: Acronyms

2 Conformance Claim

The Election administration 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 Assumptions

Assumptions	Description
A.Alert Report	Documented procedures must be implemented for responding to and reporting violations of the TOE.
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 are installed in a physically secure location that can only be accessed by authorized users.
A.Secure Installation and Operation	It is assumed that TOE and its dependencies (E.g. operating system) 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	It is assumed that the administrator are non-hostile, well trained and follow all administrator guidance.

Table 3-1: Secure usage assumptions

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.Malfunction	Users can cause malfunction like re-installation, and/or initialization of Admin 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.Recording Failure	Unauthorized access may cause audit records to be lost or modified. The unwanted outcome is that the TOE is unable to store or provide necessary audit data, or the audit becomes useless because of the inability to separate important audit records from other records.
T.Unauthorized System Modification	Unauthorized modification of the system, affecting operational capabilities, 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.Alert	TOE activity must be monitored and an auditable or visual notification must be provided to an authorized administrator.
P.Audit	Audit records are protected from unauthorized access and do not pose any security risk of the voter anonymity.
P.Least Privileges	User accounts shall be created and managed after the principle of least privilege.
P.Secure Management	Authorized administrator must manage the TOE in a secure way.
P.Test	The TOE and its associated documentation must demonstrate that it is in an accurate implementation.

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. System failures, malfunctions and other threats to the system and events at operating system level.
O.Authorization	The TOE shall provide authorization mechanisms to support minimal access privileges for the following roles: Administrator, operator and election observers. Privileges shall be the minimum necessary to satisfy the operational requirements for the Administration system of e-voting. The Administration software provides an identification and authorization procedure that creates a security token that might be used by other components and/or applications throughout the system. This token enables other components and parts of the system, that are not directly in contact with the external authentication service, to benefit from the authentication and role based access control provided by the Administrative domain.
O.Data Protection	No user data will be permanently lost in the event of TOE breakdown.
O.Digital Signature	Digital signatures are used to preserve the integrity on and verify the authenticity of files transferred between the systems.
O.Encryption	The TOE shall be able to encrypt objects exported outside the TOE and perform key destruction.
O.Integrity	Digital signatures must be used to provide integrity of data communicated between TOE and external components.
O.Manage	The TOE must provide manners that maintain the TOE secure to defined user roles of the election system. The functionality that may be requested by end-users is access to a specific role giving specific functionality.
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 to demonstrate that it is in an accurate implementation. The system software, hardware and documentation must be open for random inspection at any time.
O.TSF Data Protection	The TOE must protect TSF data from unauthorized exposure, alteration and deletion.
O.Verify	It shall be possible to verify that the TOE act accordingly to its specification. TOE must provide functionality to configure which incidents will raise an alert. The TSF shall handle the election configuration. It contains everything from parties and candidates to standard reports and reporting units

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.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 and installed, in a manner that maintains the security objectives.
OE.Management	The TOE must be managed and operated in a way that maintains security policies.
OE.Modify	Only properly authorized users shall have the possibility to delete or modify any object handled by the TOE.
OE.Non Disclosure	Disclosure of audit information to unauthorized persons shall be prevented.
OE.Physical	Appropriate physical security must be provided for the TOE.
OE.Timestamp	The TOE environment must provide reliable synchronized time sources.
OE.Trusted Administrator	Authorized administrator 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.Data Protection	O.Digital Signature	O.Encryption	O.Integrity	O.Manage	O.Self Protection	O.Test	O.TSF Data Protection	O.Verify
P.Alert	X										
P.Audit	X	X									X
P.Least Privileges							X				
P.Secure Management							X				
P.Test									X		
T.Malfunction		X						X			

	O.Audit	O.Authorization	O.Data Protection	O.Digital Signature	O.Encryption	O.Integrity	O.Manage	O.Self Protection	O.Test	O.TSF Data Protection	O.Verify
T.Management		X					X				
T.Modification of private/secret keys										X	
T.Non Integrity						X		X			
T.Recording Failure		X									
T.Unathorized System Modification		X		X	X					X	
T.Unexpected Events			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: **P.Audit** and **P.Alert.**
	O.Audit covers **P.Audit** and in a manner that TOE must audit every auditable event and collect audit records from the e-voting and p-counting domain. **P.Alert** is covered because TOE activity must be monitored and an auditable or visual notification must be provided to an authorized administrator.

O.Autorization	Threats/OSPs countered: **T.Malfunction, T.Unauthorized System Modification, T.Management, T.Recording Failure** and **P.Audit**
	The security objective **O.Authorization** cover the threats **T.Malfunction, T.Unathorized System Modification, T.Recording Failure, T.Management 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 a CAI (MinID from DIFI) or other CAIs, and an authorization policy database for role based access to securable objects.

O.Data Protection	Threats/OSPs countered: **T.Unexpected Events**
	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.Digital Signature	Threats/OSPs countered: **T.Unathorized System Modification**
	T.Unathorized System Modification is covered by the objective since Digital signatures are used to preserve the integrity on and verify the authenticity of distributed configuration files.

O.Encryption	Threats/OSPs countered: **T.Unathorized System Modification**
	T.Unathorized System Modification is covered by the objective, since TOE encryption mechanisms will protect the “security token” that provides the role based access control.

O.Integrity	Threats/OSPs countered: **T.Non Integrity**
	O.Integrity covers the threats **T.Non Integrity** in a manner that Digital signatures must be used correctly to provide integrity of data communicated between TOE and external components shall be maintained.

O.Manage	Threats/OSPs countered: **T.Management**, **P.Least Privileges** and **P.Secure Management**.
	O.Manage covers **T.Management**, **P.Secure Management** and **P.Least Privileges** in a manner that it deals with the threat of misuse TOE management functions during initialization and operation. The only way the TOE can deal with this threat is by restricting the use of TOE management functions to users authorized (**O.Authorization**) to use those functions and by auditing the actions of those users (**O.Audit**).

O.Self Protection	Threats/OSPs countered: **T.Non Integrity** and **T.Malfunction**
	O.Self Protection covers the threats **T.Non Integrity** and **T.Malfunction** since TOE are providing authorization mechanisms for protection against attempts by unauthorized users to bypass, deactivate, or tamper with security functions.

O.Test	Threats/OSPs countered: **P.Test**
	P.Test is covered by **O.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.Unathorized System Modification.**
	O.TSF Data Protection covers the threats **T.Modification of private/secret keys** and **T.Unathorized System Modification** since it states that it must protect TSF data from unauthorized exposure, alteration and deletion.

O.Verify	Threats/OSPs countered: **P.Audit**
	P.Audit is covered by **O.Verify** since it's possible to verify that the TOE act accordingly to its specification by analyzing audit data. Audit System failures, malfunctions and other threats to the system will be presented to operators.

4.4 Security Objective Rationale for the Operational Environment

4.4.1 Mapping of Operational Environment Security Objectives to assumptions

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

OE.Contigency Plan	Assumption countered: **A.Contigency Plan** and **A.Alert Report**
	A.Alert Report covers required procedures for responding to -and reporting violations of the TOE. **A.Contigency Plan** assume that there are provided a documented plan to maintain continuity of operation in an emergency or disaster.

OE.Identification and Authentication	Assumption countered: **A.Authentication** and **A.Trusted Administrator**
	OE.Identification and Authentication covers the assumptions **A.Authentication** and **A.Trusted Administrator** because it states that identification and authentication shall be effective before any action of the TOE can be carried out. This service is provided by an external CAI (MinID from DIFI) using SAML 2.0 over SOAP and HTTP.

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.Secure Installation and Operation**
	A.Secure Installation and Operation is covered by the objective **OE.Management** because it states that the TOE shall be managed in a manner that maintains the security objectives.

OE.Modify	Assumption countered: **A.Physical**, **A.Authentication**, **A.Secure Installation and Operation** and **A.Trusted Administrator**
	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**) 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**
	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.

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

5.1.1.1 Management of security functions behavior (FMT_MOF.1)

FMT_MOF.1.1 The TSF shall restrict the ability to [*modify the behavior of*] the functions [*TSF management functions (*FMT_SMF.1*)*] to [*System administrators*].

5.1.1.2 Management of security attributes (FMT_MSA.1)

FMT_MSA.1.1 The TSF shall enforce the [*Based Access Control (RBAC)*] to restrict the ability to [modify] the security attributes [*all TSF and user data protected by the TOE*] to [*System administrator*].

5.1.1.3 Specification of Management Functions (FMT_SMF.1)

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions: [*User/role management (Authorization), Security Audit generation and management, Encryption and keys for signing, Election configuration and administration*].

5.1.1.4 Secure security attributes (FMT_SMR.1)

FMT_SMR.1.1 The TSF shall maintain the roles [*System administrator, Auditor, Observer and application users*].

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

5.1.2 Identification and authentication (FIA)

5.1.2.1 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.

5.1.2.2 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.

5.1.3 Security Audit (FAU)

5.1.3.1 Security Alarms (FAU_ARP.1)

FAU_ARP.1.1 The TSF shall take [*Generate and send alarm*] upon detection of a potential security violation.

5.1.3.2 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 operation of the election system and its communications infrastructure, 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*].

5.1.3.3 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.

5.1.3.4 Potential violation analysis (FAU_SAA.1)

FAU_SAA.1.1 The TSF shall be able to apply a set of rules in monitoring the audited events and based upon these rules indicate a potential violation of the enforcement of the SFRs.

FAU_SAA.1.2 The TSF shall enforce the following rules for monitoring audited events:

a) Accumulation or combination of [*Illegal access, Integrity errors, modification of TSF configuration data and unavailability of resources audited*] known to indicate a potential security violation;

b) [*none*].

5.1.3.5 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.

5.1.3.6 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.

5.1.4 Protection of the TSF (FPT)

5.1.4.1 Inter-TSF detection and correction 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.

5.1.4.2 Reliable time stamp (FPT_STM.1)

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

5.1.5 Cryptographic support (FCS)

5.1.5.1 Cryptographic key generation (FCS_CKM.1)

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [*AES*] and specified cryptographic key sizes [*256 bits*] that meet the following: [*FIPS 197*].

5.1.5.2 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*].

5.1.5.3 Cryptographic operation (FCS_COP.1)

FCS_COP.1.1 The TSF shall perform [*encryption of "security token" exported to the p-voting domain*] in accordance with a specified cryptographic algorithm [*AES*] and cryptographic key sizes [*256 bits*] that meet the following: [*FIPS 197*].

5.1.6 User Data Protection (FDP)

5.1.6.1 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.

5.1.6.2 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*].

5.1.6.3 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, Auditor 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.

5.1.6.4 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 and configuration data 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 or in case of hardware, software and/or storage devices fault*].

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

5.2 Security Functional Requirements dependencies

Requirement	Dependencies	Dependency met
FMT_MOF.1	FMT_SMR.1 and FMT_SMF.1	Yes
FMT_MSA.1	FMT_SMR.1, FMT_SMF.1 and FDP_ACC.1	No
FMT_SMF.1	None	Yes
FMT_SMR.1	FIA_UID.1	Yes
FIA_UID.2	None	Yes
FIA_UAU.2	FIA_UID.1	Yes
FAU_ARP.1	FAU_SAA.1	Yes
FAU_GEN.1	FPT_STM.1	Yes
FAU_GEN.2	FAU_GEN.1 and FIA_UID.1	Yes
FAU_SAA.1	FAU_GEN.1	Yes
FAU_SAR.1	FAU_GEN.1	Yes
FAU_SAR.2	FAU_SAR.1	Yes
FPT_ITI.1	None	Yes
FPT_STM.1	None	Yes
Feil! Fant ikke referansekilden.	FCS_COP.1 and FCS_CKM.4	Yes
FCS_CKM.4	FCS_CKM.1	Yes
FCS_COP.1	FCS_CKM.1 and FCS_CKM.4	Yes
FDP_ACC.2	FDP_ACF.1	Yes
FDP_ACF.1	FDP_ACC.1 and FMT_MSA.3	No
FDP_DAU.2	FIA_UID.1	Yes
Feil! Fant ikke referansekilden.	FDP_ACC.1	No

Table 5-1: Security Functional Requirements dependencies

5.2.1 Justification why dependencies are not met

FMT_MSA.1 has dependency to FDP_ACC.1 which is not defined as a requirement for this TOE. FDP_ACC.2 is covering the dependency since it requires TSF to enforce the access control among subjects and objects.

FDP_ACF.1 has dependency to FDP_ACC.1 which is not defined as a requirement for this TOE. FDP_ACC.2 is covering the dependency since it requires TSF to enforce the access control among subjects and objects. FMT_MSA.3 is also listed as a required dependency but Static attribute initialization for default values of security attributes is not required for this TOE. FMT_MSA.1 cover management of security attributes allows authorized users (roles) to manage the specified security attributes; this will cover required dependency for FDP_ACF.1.

FDP_ROL.1 has dependency to FDP_ACC.1 which is not defined as a requirement for this TOE. FDP_ACC.2 is covering the dependency since it requires TSF to enforce the access control among subjects and objects.

5.3 Mapping of TOE Security Requirements to Objectives

The following table shows the mapping between TOE Security Requirements and Objectives defined in this ST.

	Feil! Fant ikke	**Feil! Fant ikke**	O.Data Protection	O.Digital Signature	O.Encryption	O.Integrity	**Feil! Fant ikke**	**Feil! Fant ikke**	**Feil! Fant ikke**	**Feil! Fant ikke**	**Feil! Fant ikke**
FMT_MOF.1		X					X	X		X	
FMT_MSA.1		X					X			X	
FMT_SMF.1		X					X			X	
FMT_SMR.1		X					X				
FIA_UID.2								X			
FIA_UAU.2								X			
FAU_ARP.1	X								X		
FAU_GEN.1	X							X	X		
FAU_GEN.2	X							X			
FAU_SAA.1	X								X		X
FAU_SAR.1	X	X									
FAU_SAR.2		X									
FPT_ITI.1				X		X					X
FPT_STM.1	X										
FCS_CKM.1					X						
Feil! Fant ikke referansekilden.					X						
Feil! Fant ikke referansekilden.					X						
FDP_ACC.2		X									
FDP_ACF.1		X									
FDP_DAU.2						X					
FDP_ROL.1			X								

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

5.4 Justification of TOE Security Requirements to Objectives

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

FMT_MOF.1	Objectives addressed: **O.Authorization, O.Manage**, **O.TSF Data Protection** and **O.Self Protection**
	FMT_MOF.1 requires the TSF to restrict the ability to modify the management functions of the TOE to authorized users only. This requirement covers the objective **O.Authorization** since privileges shall be the minimum necessary to satisfy the operational requirements for the Administration system of e-voting. **O.Manage**, **O.TSF Data Protection** and **O.Self Protection** are covered by this objective since TOE must protect itself against operations performed by unauthorized users.

FMT_MSA.1	Objectives addressed: **O.Authorization**, **O.Manage** and **O.TSF Data Protection**
	FMT_MSA.1 requires the TOE to provide a tool that can be used by authorized users to manage access control SFPs. Objectives regarding authorization, management – and TSF data protection will be covered by this requirement.

FMT_SMF.1	Objectives addressed: **O.Authorization**, **O.Manage** and **O.TSF Data Protection**
	FMT_SMF.1 requires the TOE to perform management functions in order to cover objectives regarding TSF data protection, configuration and administration.

FMT_SMR.1	Objectives addressed: **O.Authorization** and **O.Manage**
	FMT_SMR.1 requires the TOE to be able to maintain the different user roles defined in the system as a part of the authorization mechanisms provided by the TOE. This will cover the objectives **O.Authorization** and **O.Manage.**

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.**

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.**

FAU_ARP.1	Objectives addressed: **O.Test** and **O.Audit**
	FAU_ARP.1 requires TSF to generate and send alarm upon detection of a potential security violation. This requirement cover the objective for audit as defined in **O.Audit** and contribute with possibilities for inspections as described in **O.Test**.

FAU_GEN.1	Objectives addressed: **O.Audit**, **O.Self Protection** and **O.Test**
	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.Test**. It also ensures that self protection is provided in a matter of performing corrective actions (**O.Self Protection**).

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**).

FAU_SAA.1	Objectives addressed: **O.Verify**, **O.Test** and **O.Audit**
	FAU_SAA.1 requires that the TSF shall be able to apply a set of rules in monitoring the audited events and based upon these rules indicate a potential violation of the enforcement of the SFRs (**O.Verify** and **O.Test)**. Logs, election configuration and verification data will be recorded to make this available for audit functions performed by the TOE as described in **O.Audit**.

FAU_SAR.1	Objectives addressed: **O.Authorization** and **O.Audit**
	FAU_SAR.1 Requires TSF to provide generated audit data to authorized user roles only which covers **O.Authorization**. Further TSF shall provide the audit records in a manner suitable for the user to interpret the information which covers **O.Audit**.

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**.

FPT_ITI.1	Objectives addressed:**O.Digital Signature**, **O.Verify** and **O.Integrity**
	FPT_ITI.1 requires that the TSF shall provide the capability to detect modification of all TSF data (**O.Verify**) during transmission between the TSF and another trusted IT product. Modifications detected by a digital signature cover the objective **Feil! Fant ikke referansekilden.** and **O.Digital Signature**.

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.

FCS_CKM.1	Objectives addressed: **O.Encryption**
	FCS_CKM.1 requires the TSF to generate cryptographic keys and therefore cover the objective **O.Encryption**. Admin will generate a key to encrypt the "security token" for authorization used in the p-voting domain.

Feil! Fant ikke referansekilden.	Objectives addressed: **O.Encryption**
	Key destruction as described in **O.Encryption** is performed by mechanisms as described in **Feil! Fant ikke referansekilden..**

FCS_COP.1	Objectives addressed: **O.Encryption**
	"Security token" for RBAC shall be encrypted when transferred between systems in accordance with the cryptographic algorithm as described in **FCS_COP.1** to satisfy **O.Encryption.**

FDP_ACC.2	Objectives addressed: **O.Authorization**
	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**.

FDP_ACF.1	Objectives addressed: **O.Authorization**
	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**.

FDP_DAU.2	Objectives addressed: **O.Integrity**
	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.

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.

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
- Security management (Election configuration and administration)

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	Security management
FMT_MOF.1				X
FMT_MSA.1				X
FMT_SMF.1				X
FMT_SMR.1				X
FIA_UID.2			X	
FIA_UAU.2			X	
FAU_ARP.1	X			
FAU_GEN.1	X			
FAU_GEN.2	X			
FAU_SAA.1	X			
FAU_SAR.1	X			
FAU_SAR.2	X			
FPT_ITI.1				X
FPT_STM.1				X
FCS_CKM.1		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

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 system logs all significant events, recording among others user, time and event details, 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

The TOE will be able to generate keys and use AES algorithm to encrypt the “security token” used in the p-voting domain. This key will be generated based on a user provided password. The TSF shall also be able to overwrite these cryptographic keys.

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.
- The modification authorization procedure.
  - If the role-owner may add other users to the role on his own, or if collaboration with a System Operator is needed.
  - Which other roles must be involved when adding or removing access to the securable objects. (The role owner can never add securable objects on his own).

6.1.4 Security management (Election configuration and administration)

The Administration software will handle the election configuration. This includes management of users and roles (Authorization), security audit generation and management, encryption and keys for signing, election configuration and administration.

The user will be able to set attribute values describing the characteristics of an election. All attributes that describe an election 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. Party codes (and party names) can be imported from different sources.

An administrator can configure the necessary number of levels, e.g. covering voting districts (bydel) or limited geographical areas for referendums. Normally it will be three configuration levels for the elections:

1. Central level
2. Local level-1 (county)
3. Local level-2 (municipality)

All previous election configuration templates will be available for editing or as a basis for creating a new configuration. An election can be configured 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.