ErgoGroup

E-vote 2011
Security Architecture Description
eVoting TOE

V 1.0

Change history

Date	Version	Description	Author
26.11.2010	0.1	Initial.	MARHAR
01.12.2010	0.2	Divided per TOE.	SCYTL R&D
03.12.2011	0.5	Draft Version.	SCYTL R&D
31.05.2011	0.6	Content is updated regarding infrastructure changes.	SCYTL R&D
08.06.2011	0.7	First full Version.	SCYTL R&D
09.06.2011	0.8	Initialisation processes are updated.	SCYTL R&D
03.06.2011	0.9	Added disclaimer, and copyright and next version target	SCYTL
09.06.2011	1.0	Review	SCYTL R&D

DISCLAIMER: Some information in it this document might be obsolete, inaccurate or might be missing. Updates will be made if such discrepancies are found. This disclaimer will be also updated to reflect the state of the document.

NEXT VERSION TARGET DATE:

Version	Date	Author	Comments

Approval

Approved by	Role	Sign	Date
Svein Endresen	Project Manager		22.03.2011
Svein Winje	Technical Architect		22.03.2011
Dan Sørensen	Customer		

Contents

1. Introduction 5

2. Overall conceptual design 6

2.1 Main security domains and the interfaces between them 6
2.2 Main user groups 8
2.3 Authentication 8
2.4 Authorisation and access control 10

3. Overall logical design 12

3.1 Application architecture 12
3.2 Conceptual security design – eVoting 13
3.3 Logging, auditing, and monitoring 15

4. Overall physical design 15

4.1 Desktop environment 15
4.2 Local area network environment 15
4.3 Wide area networking environment 15
4.4 Solution topology 16

5. Secure initialisation process 17

6. Domain isolation 19

6.1 Physical Isolation. 19
6.2 Logical Isolation. 19

7. Self-protection 20

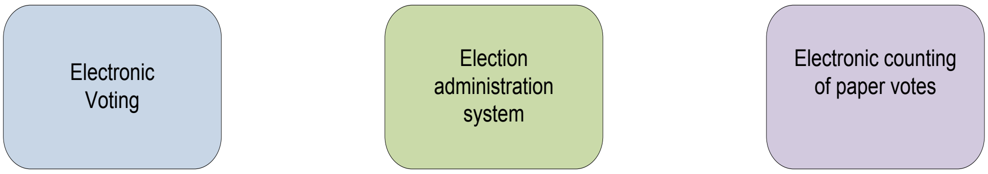
7.1 Configuration protection. 20
7.2 Software protection. 20
7.3 Segregation of duties protection. 21

8. Non-bypassibility 22

1. Introduction

This document presents the security architecture of the election systems which is part of the E-vote 2011 project. By *security architecture*, we mean the conceptual, logical, and physical design which is used to protect the systems and the information they protect from compromise.

The E-Vote 2011solution consists of three different but interacting systems:


Electronic Voting
Election administration system
Electronic counting of paper votes

Each system can operate in its own environment, but require interaction to satisfy common requirements such as configuration, authentication/authorization, logging and reporting. The overall security architecture is described in the document “Security Architecture - General Overview”.

2. Overall conceptual design

Conceptual Design is the view of the security solution *from the user point of view*. It seeks to define the characteristics of information access and authentication that are part of the user experience.

2.1 Main security domains and the interfaces between them

The electronic voting domain includes three separate domains as illustrated below.

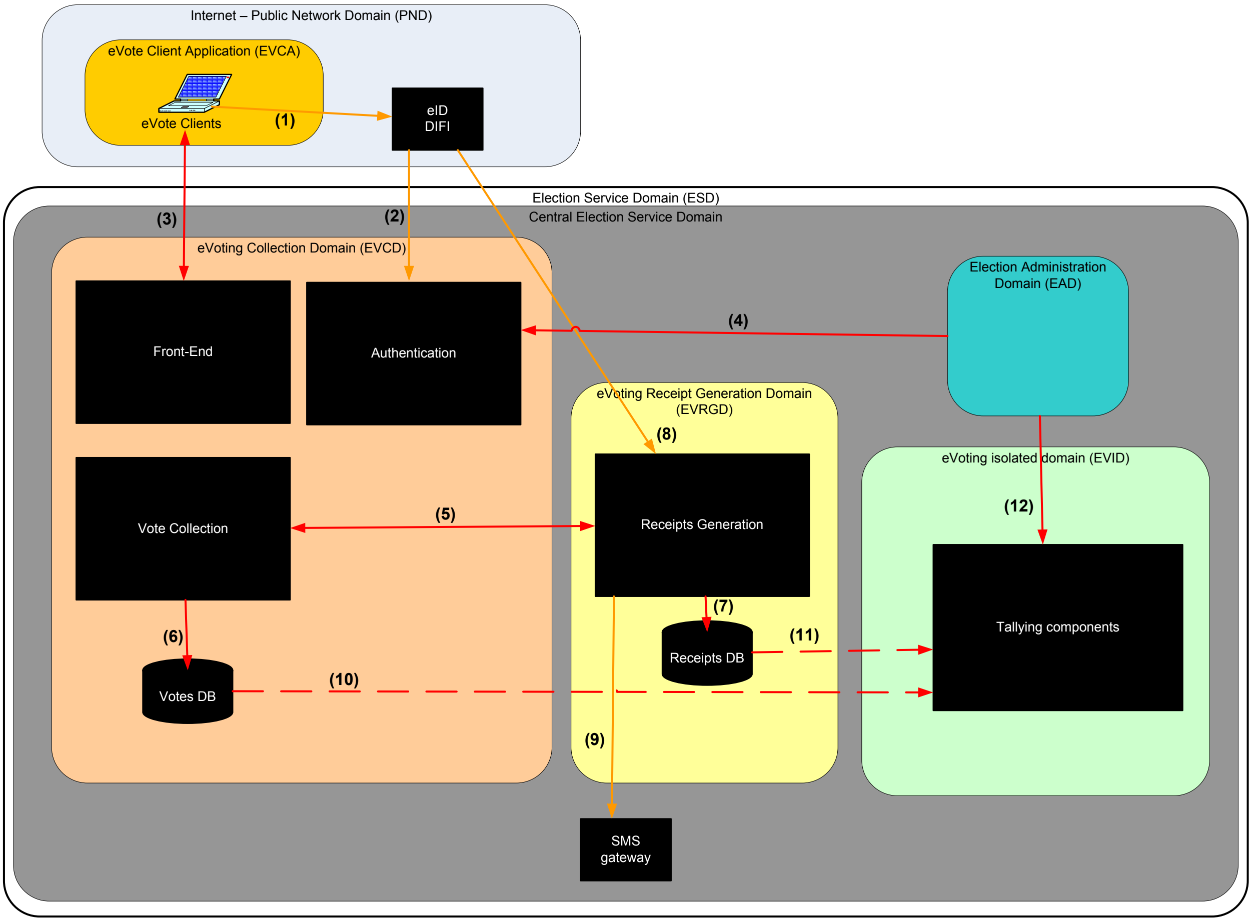

Internet – Public Network Domain (PND)
eVote Client Application (EVCA)
eVote Clients
(1)
eID
DIFI
Election Service Domain (ESD)
Central Election Service Domain
(3)
(2)
eVoting Collection Domain (EVCD)
Front-End
Authentication
(4)
Election Administration Domain (EAD)
eVoting Receipt Generation Domain (EVRGD)
(8)
eVoting isolated domain (EVID)
(12)
Vote Collection
(5)
Receipts Generation
Tallying components
(7)
Receipts DB
(11)
(6)
Votes DB
(10)
(9)
SMS gateway

Figure 1: The eVotingDomain (EVD)

- The **e-Voting Collection Domain (EVCD)** contains the IT infrastructure to host the voter authentication process and the main part of the electronic ballot collection process. Main components from this domain are the Authentication Service (AS) which ensures the voters authentication and eligibility, and the Vote Collector Server (VCS) which receives, verifies, and process the votes, and stores them in the Ballot Box.
- The **e-Voting Receipt Generation Domain (EVRGD)** contains the components which are collaborating with the EVCD components in the voting process. The main component from this domain is the Return Code Generator (RCG) which mission is to verify the votes and create voting receipts and return codes for voter verifiability. This domain is separated from the EVCD and only communicated with a controlled VPN connection, due to security reasons.
- The **e-Voting isolated Domain (EVID)** contains the components which are going to execute the tallying operations (Cleansing, Mixing, and Counting). Since these components are performing the most critical operations (cleansing, mixing, decrypting, and counting) they are isolated from any network, and operated in a controlled environment.

The required interactions between the systems are done through the following interfaces (numbered in the figure above):

1. **User identification**. Web interface over HTTPS, redirected from the eVoting front-end. Described in the section "Authentication in general" below.
2. **Verify user authentication**. Web service over authenticated HTTPS connection. Described in the section "Authentication in general" below.
3. **Vote casting** over HTTPS. The voter is sending her selected voting options, encrypted and digitally signed. The voting client receipts vote casting confirmation, and a voting receipt to verify the correct vote processing.
4. **Electoral Roll validation**. Check if a voter is eligible to cast a vote in an election with a web service. Described in the section "Authentication in general" below.
5. **Voting receipt request**. The VCS send the vote to the RCG, for vote verification and generation of voting receipts and return codes.
6. **Ballot Box**. All valid votes verified by VCS and RCG are stored in the ballot box database.
7. **Voting Receipts – DB.** The voting receipts generated by the RCG are stored; they will be compared in the tallying process for validating the stored votes.
8. **SMS messages**. The return codes generated by the RCG component are sent to the voters through a SMS gateway.
9. **Ballot Box export**. All votes from the Ballot Box database are exported to the e-Voting isolated Domain (EVID). It is a manual and air-gap interface.
10. **Voting receipts export**. All voting receipts are exported to the e-Voting isolated Domain (EVID). It is a manual and air-gap interface.
11. **Voter phone number request**. The RCG component needs the voter phone number to send her an SMS message with her correspondent return codes. This phone number is obtained by requesting the IDPorten service through an encrypted and mutual authenticated connection.
12. **Electoral Roll and markoffs**. Get a record based file for the whole electoral roll with mark-offs. The electoral roll copy is used to check the final electronic ballots and to filter out all voters which have cast a paper vote from the final electronic count. It is a manual and air-gap interface.
13. **Counting of eVotes**. Deliver eVoting count results in EML (XML) format with the same web service as 5b over a mutual authenticated HTTPS connection. The counting EML files from eVoting are digitally signed.
14. An additional interface – not included in the diagram – exists for all the eVoting components. The deliver copy of local **log events** through the same rsyslog service as 5c (UDP/514). The audit log data is immutabilized before it is sent.

More detailed information on the interfaces could be found in System Architecture documents.

2.2 Main user groups

The systems have several different user groups which are described in more detail under each. The main user groups are:

1. **Voters:** authenticate and cast their electronic vote in a public web application. Only eligible voters in the election roll could cast their vote.
2. **Election officials / operators**: setup and configure the eVoting components, both online and air-gap servers. Also report the counting results from the electronic voting.
3. **Electoral Board**. Group responsible for protecting the voters' privacy. They are sharing the splits from the cryptographic election private key which is used for decrypting the votes before vote counting.
4. **Administration Board**. Group responsible for the election configuration. They have to verify and digitally sign the election configuration information, and the counting results.
5. **Auditors.** External auditors shall participate for validating all the election process. Most complex operations like the mixing and decrypting processes are generating specific evidences for auditors.
6. **Operator / monitoring.** Election Monitoring Profile. They can only executed read-only operations which are not accessing any private information (accessing logs, and executing application tests).

User role	General use
Voters	Authentication and vote casting
Election Officials / operators	Election configuration (online servers)
	Election configuration (air-gap servers)
	Tallying Operations (online servers)
	Tallying Operations (air-gap servers)
Administration Board	Digital signature of Election configuration
	Digital signature of Counting results
Electoral Board member	Votes decrypting
Auditor	Election auditing (data integrity, mixing operation, decrypting, cleansing activities, results…)
Operator / monitoring	Application status monitoring
	Application Logs monitoring

2.3 Authentication

2.3.1 Voters Authentication

Voters are authenticated through MinID portal. This authentication mechanism is described in the document "Security Architecture - General Overview". This is in principle a single-sign-on system for many government services, but the configuration does not allow single-sign on towards the election systems.

Voters can be also authenticated in the polling-places by means of her National ID card. In the same way as explained in the "Security Architecture - General Overview", the application in the poll-sites is generating identification tokens in the same way than the MinID service (SAML assertion).

Accessing from the Internet – Public Network Domain (PND) cannot be controlled. The voting terminals used by voters download the Voting Client – applet used for eVoting service, but neither the voter PC nor the applet

application can be authenticated.
When accessing from the controlled environment, the voting terminals are further authenticated based on the calling IP address and provided SSL client certificate. Before use of the eVoting client, the client certificate must be installed by municipal system administrator. The voting terminal should be configured in a secure way, be well patched, and have an updated anti-virus system running.

Inactivity session timeout in eVoting is configurable. It is validated by the VCS when storing the vote; since we cannot trust on the date and time provided by the voter PC, it is validated by comparing the authentication timestamp with the receipt generation timestamp. By default, it will be set to be 30 min.

2.3.2 Users Authentication

Any other user except voters shall be authenticated through RBAC access control – which is outside the eVoting TOE but in the Admin TOE.
Any user successfully authenticated in RBAC by means of user_id and password, will obtain an RBAC Token containing the rights this user is granted for. This RBAC Token is digitally signed by the RBAC service, encrypted with an AES 128 symmetric key, and should be stored in any media to be provided to the eVoting application.

A valid RBAC token will contain the following fields:

- A unique ID.
- The operator name
- The election event
- Creation and expiration date and time.
- Role information (name, id, context…)
- Accesses.

The eVoting just needs to verify the RBAC authentication token by means of:

- Requesting the token encryption password to the user. This password shall be used to decrypt the token.
- Verifying the RBAC token digital signature.
- Verifying the validity of the RBAC digital certificate used for verifying the signature.
- Verify the election event is
- Verify the creation time is not in the future.
- Verify the expiration time is not in the past.

As mentioned, this authentication token expires when the “expiration” field inside the token indicates.

All authentication operations regarding RBAC tokens are registered at the application immutable logs.

2.4 Authorisation and access control

There are two different authorisation processes:

- Voter authorisations for eVoting.
- Election Officials / operators, Mixing Auditors, and Electoral Board members.

2.4.1 Voter authorisations for eVoting.

The only authorisation the eVoting system shall consider for voters is the voter eligibility.
The voter authorisation and access control for voters in the eVoting control can be divided in two steps:

- Obtaining the authorisation for the voters.
- Verifying the voter authorisations for casting a vote.

Obtaining the authorisation for the voters
After the authentication process – performed by MinID or by the electoral officials - the Authentication Server obtains the contests the voter is eligible to vote, by means of:

- Receives an authentication token (from MinID or from the poll-sites).
- Verifies the information contained in the authentication token (digital signature, digital certificates, unique_id, and expiration time).
- Extracts the voter_id from the identification token.
- Sends this voter_id to the "Electoral Roll Service" in the Admin TOE, requesting a message confirming that the voter is included in the Electoral Roll, and what contests are eligible to vote.

Once the authentication server has received the contests the voter is eligible to vote from the Electoral Roll, it creates a new Authentication token containing:

- A unique ID.
- The MinID or poll-site authentication token.
- The voter_ID.
- Creation and expiration dates and times.
- The contests the voter is eligible to vote.

This Authentication Token is digitally signed by the Authentication Service.

Verifying the authorisation for the voters

The verification of the voters' authorisations is performed three times:

- In the VCS when receives a vote from the voter.
- In the RCG when receives a vote from the VCS.
- In the Cleansing component when verifying the whole ballot box.

The authorisation verifications are the following:

- The Min_ID or poll-site authentication token is verified (digital signature, digital certificates, unique_id, and expiration time).
- The authentication token generated by the Authentication Service is verified (digital signature, digital certificates, unique_id, and expiration time).
- The digital certificates used for verifying the digital signature of the vote, shall be referred to the voter_id specified in the Min_ID or poll-site authentication token, and in the Authentication Token from the Authentication Service.
- The vote shall be referred to a contest the voter is eligible to vote, according to the contests specified in

the Authentication Token from the Authentication Service.
During the Cleansing process, this is also verified against an “Electoral Roll” list which is digitally signed by the Admin System.

All voter authentication operations are registered at the application immutable logs.

2.4.2 Election Officials / operators, Mixing Auditors, and Electoral Board members.

The eVoting system defines a set of “securable objects” which are the different objects managed by the eVoting application which can be interacting with the user.
All the eVoting securable objects have been analyzed, considering which actions an specific user would request to perform, which ones shall be performed only by the application – with no user interaction, and which ones shall not be performed by nobody.

By this way, a list of actions which could be executed by a user have been defined. These actions can be summarized in:

- Upload configuration files to the eVoting components (voter credentials, EML file, public params, areas, key stores, and return codes).
- Export the Ballot Box from the VCS.
- Export the Voting Receipt List from the RCG.
- Import the information in the Cleansing (the Ballot Box, the Voting Receipts, the markoffs, the electoral roll).
- Perform the Mixing operations (import the cleansed Ballot Box, execute the mixing, export the mixed ballot box).
- Perform the mixing audit activities (introduce random, validate proofs).
- Perform the counting activities (import the mixed ballot box, export the counts)
- Request votes decryption.

Any of these operations are implement to request an RBAC token to grant access. Once the user provides the RBAC token, and it is validated (as explained in the authentication chapter), the application verifies that the string related to the requested action is included in the “accesses” tag from the RBAC token. Otherwise, the action shall be rejected.

All user authentication operations are registered at the application immutable logs.

3. Overall logical design

3.1 Application architecture

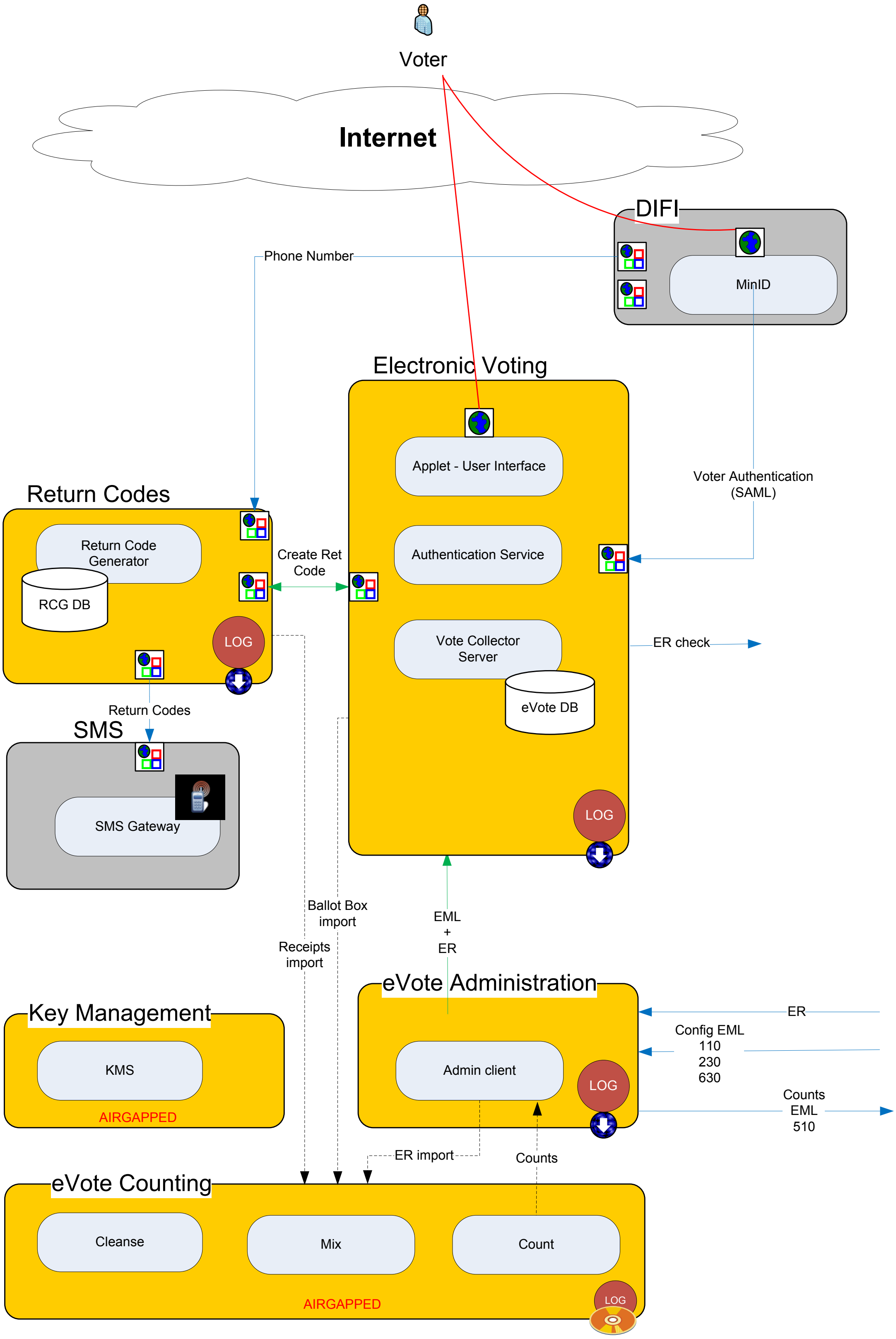


Figure 2: Overall logical design

3.2 Conceptual security design – eVoting

The eVoting module consists of 5 main Security Functions:

- Identification and authorization
- Process Integrity and Accuracy
- Cryptographic support
- Security audit
- Service Availability

The following rationale explains how the software is providing its Security functions.

Identification and authorization

It is mentioned in a previous chapter of this document.

Process integrity and Accuracy

The Electronic Voting Software will be responsible for the vote casting, safe storing, processing and counting. Because the high importance of these processes, the software shall ensure that there is no possibility to perform any unexpected or non-authorized operation over the voting process.
Voter confidentiality, and polling integrity and confidentiality, shall be protected.
Three information flows have been defined to manage and protect the security of the process:

1) **Election Management**: Election configuration management, electoral period opening and closing process, electoral administration activities, cryptographic keys generation and management, and voting cards generation.
2) **Voting Process:** Voter Identification and Authentication, vote casting, vote checking, vote receipt generation and sending, return code generation and sending, vote storing.
3) **Tallying Process.** Starting the tallying process, cleansing process, mixing process (includes auditing), counting process, determining the electoral result.

Additional security controls have been defined to protect the communications operations and the data integrity.

Cryptographic support

Every critical information asset and operation will be protected through cryptographic mechanisms which ensure information confidentiality, authenticity, and integrity. E.g:

- Communications are encrypted between the PC of the voters and the Front End.
- Votes are encrypted and digitally signed by the voter.
- Election Configurations are digitally signed by the electoral administrators.
- The ballot box is digitally signed by every service which manipulates it (Vote Collector Service, Cleansing Service, Mixing Service, and Counting Service).
- Vote receipts are digitally signed by the Return Code Generator.
- Log Files are digitally signed by each component of the Electronic Voting Software.
- …

The Electronic Voting Software will use cryptographic keys and algorithms in accordance with the following standards:

- **Key generation algorithm**: RSA, minimum key size equivalent to a symmetric key of 100 bits (FNISA (French Network and Information Security Agency) Recommendations 2010); meet the specifications in FIPS 186-3.
- **Digital signature and verification**: RSA, minimum key size equivalent to a symmetric key of 100 bits; specifications in PKCS#1 v2.1 for the RSA digital signature and verification (RSASSA-PSS);
- **Hash algorithm**: SHA-256; specifications in FIPS 180-3 ;
- **Message authentication**: HMAC, minimum key size 128 bits; specifications in FIPS 198a for HMAC function in combination with SHA-256 hash function.

Security audit

All components from the eVoting platform send a copy of their logs to the audit system. If a system or infrastructure component is broken, a local copy of the log is always available.

The system logs all significant events, recording the user, time, and event details. This includes logs of all events at all levels of the complete Electronic Voting Software. It also includes all voting transactions, attacks on the operation of the electronic voting system, its system failures, malfunctions and other threats to the system and events.

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:

- Date and time of the event.
- Type of event.
- Related object identification.
- Subject identity (username, session id, IP address and other location information from the user who generates the event).
- And the outcome (success or failure) of the event.

Logs are stored with the “Immutable log” mechanism, which ensures that any change to the log files would be detected.

Service Availability

The availability of the Electronic Voting Service shall be ensured, in order to allow the voters to exercise his right-to-vote.

The following security measures will be implemented to ensure the availability of the eVoting Service:

- A high-availability infrastructure, with several replicated servers for each functions.
- Two different datacenters per location (two per Bronosuynd, and two per DSB) containing replicated infrastructures.
- Load-Balancers to distribute the voters to the different servers and datacenters.
- When the eVoting services are starting, every component is performing several self-status verifications to ensure that the component can be started.
- Service-checks will be available to the system operators, to verify the status of the service.
- All the components of eVoting will have the capacity to be activated or deactivated if they are experiencing problems.
- Critical operations will have the property that the function either completes successfully, or for the indicated failure scenarios, recovers to a consistent and secure state.

3.3 Logging, auditing, and monitoring

The central log and audit is a separate system which collects log events from eVoting and the other election systems. It is part from the Admin system.

A detailed study has been performed to analyze:
- What information should be registered in the logs.
- What events should be registered in the logs by the eVoting applications to ensure traceability.
- What errors or anomalous behaviours should be registered at logs by the eVoting application.
- What alarms shall be defined in the monitoring system related to the eVoting application logs.

The results of this study have been executed by implementing the required log entries in the eVoting applications, and the required security alerts in the SPLUNK monitoring system.

All the detailed information for logging, auditing, and monitoring, could be found in the audit architecture, and logs and alerts design documentation.

4. Overall physical design

Physical Design is the view of the security solution from the developer's point of view. It seeks to define the configuration of the physical components used to implement the technology solution and to provide a process for operation and support of the system.

4.1 Desktop environment

We have no influence on the clients used by the electronic voters or any other user. Voters could use any standard web browser on any platform. Any other user accessing the eVoting platform could use her PC on any platform.

4.2 Local area network environment

The systems for the e-Voting Collection Domain are deployed at Brønnøysund service provider, and the systems for the e-Voting Receipt Generation Domain (EVRGD) are deployed at DSB service provider.
Both are deployed in their own local area networks, but a VPN is deployed to join a specific web service from the VCS and RCG. This VPN is restricted and controlled to ensure only allowed operations are performed.

The systems from the e-Voting isolated Domain (EVID) are isolated from any network, so they are not connected to any LAN.

4.3 Wide area networking environment

The web application for electronic voting are accessed from internet. SSL server certificate are installed and all connections are by HTTPS.

Neither the systems from the -Voting Receipt Generation Domain (EVRGD) nor the e-Voting isolated Domain (EVID) are accessible through any WAN.

4.4 Solution topology

The central part of the E-voting and Admin is deployed in two data centres at Brønnøysund. The system needed for handling the return codes for E-voting are deployed in two data centres at Directorate for Civil Protection and Emergency Planning (DSB). The offline (airgapped) systems used for cleansing and mixing of E-votes are placed in KRD's own data centres.

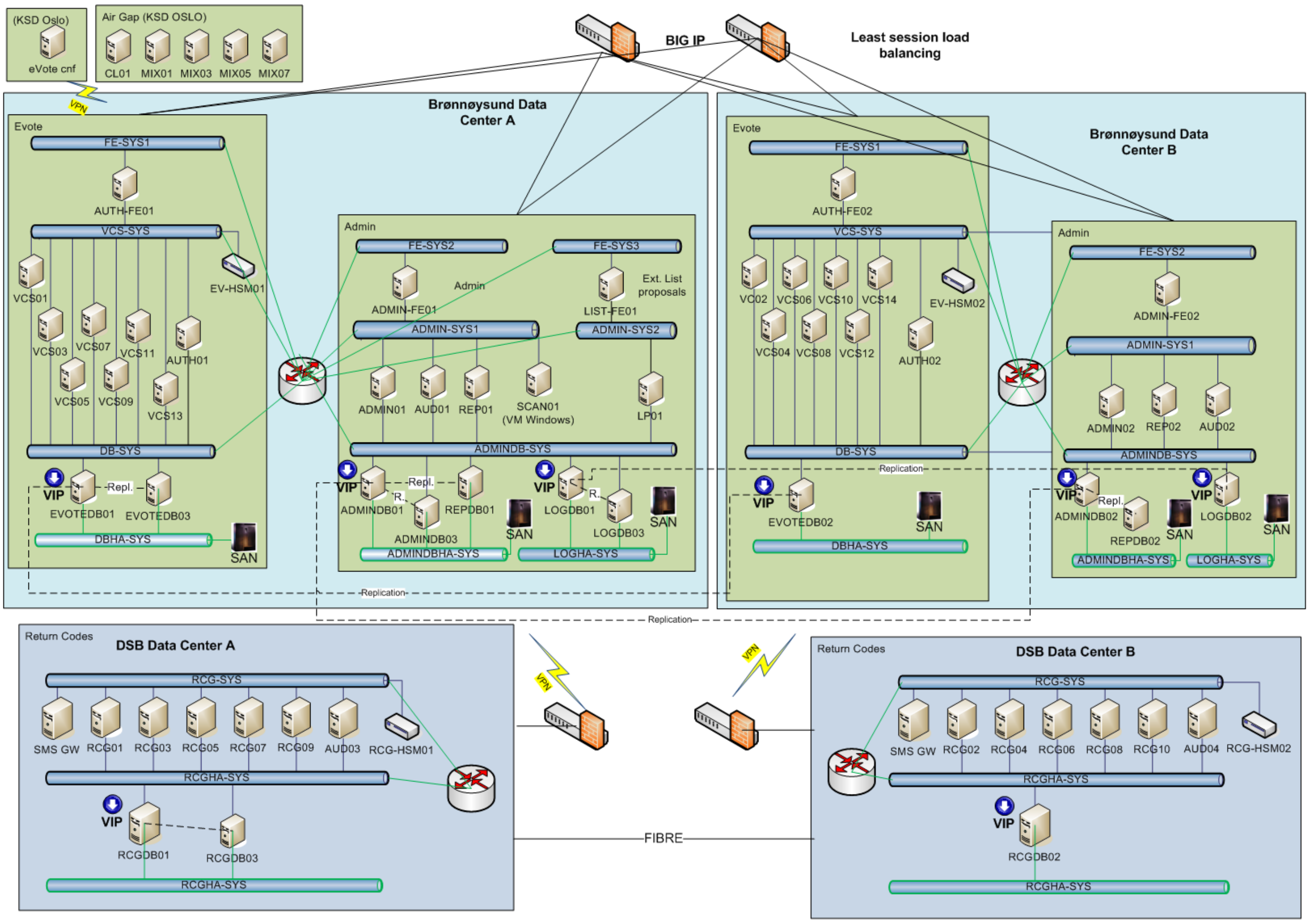


Figure 3: Operational environment

5. Secure initialisation process

On starting the eVoting service (the online components) the system shall perform the following security checks to ensure the system is properly configured, secured, and initialised, before activating the voting service:

1. **Self-test of software functionality and information integrity:**

- **VCS.**

  i. Prints bootstrap status (found files, keystores and rbac certificates for many elections).
  ii. Checks that the RBAC certificate is present for the election.
  iii. Signs & verifies with the VCS rsa key (HSM or PKCS12)
  iv. Obtains the VCS elgamal private key (Tests also decryption with the PKCS12).
  v. Obtains the VCS elgamal symmetric key (Tests also decryption with the PKCS12).
  vi. Verifies the EML signature and that the content of the database is coherent with the EML.
  vii. Get ballots number from DB.
  viii. Checks audit logger service activated

- **RCG.**

  i. Prints bootstrap status (found files, keystores and rbac certificates for many elections).
  ii. Checks that the RBAC certificate is present for the election.
  iii. Signs & verifies with the RCG rsa key (HSM or PKCS12)
  iv. Obtains the RCG elgamal private key (Tests also decryption with the PKCS12).
  v. Obtains the RCG elgamal symmetric key (Tests also decryption with the PKCS12).
  vi. Checks the phone service connection (retrives status for a ssn)
  vii. Checks the SMS gateway.
  viii. Verifies the EML signature and that the content of the database is coherent with the EML.
  ix. Checks the RCG service (DB, number of return codes).
  x. Checks the RCG receipt service (DB, number of voting receipts)
  xi. Checks audit logger service activated

- **AS.**

  i. Prints bootstrap status (found files, keystores and rbac certificates for many elections).
  ii. Checks that the RBAC certificate is present for the election.
  iii. Signs & verifies with the RCG rsa key (HSM or PKCS12)
  iv. Checks the phone service connection (retrieves phone number for a ssn)
  v. Checks the IDP connectivity.
  vi. Checks the electoral roll service (retrieves information for a ssn)
  vii. Checks audit logger service activated

2. Cryptographic Keys verification.

The following verifications are being performed once for a set of AS-VCS-RCG servers:

- **Cryptographic keys for signature are properly generated, distributed, and stored.**

  i. The AS, VCS, and RCG are able to digitally sign.
  ii. The VCS is able to verify what the AS and RCG are digitally signing.
  iii. The RCG is able to verify what the AS and VCS are digitally signing.

- **Cryptographic keys for decryption are properly generated, distributed, and stored.**
  i. A message with the vote's structure is generated, using a non-existent voter.
  ii. This message is encrypted with the election public key.
  iii. This message is processed by the VCS, which performs the partial decryption using its private key
  iv. This partially-decrypted message is sent to the RCG, which tries to decrypt it using its private key.
  v. The RCG tries to obtain the return codes to verify its correct generation.
  vi. Since it is only a test-vote, it is not processed or stored like normal votes.

When the processes are completed successfully, a baseline is created with a hash of the 3 keystores (AS, VSC, RCG). This baseline is used to verify the keystores of the other servers.

3. Verification of the connectivity between components:

All the following components are sending a verification messages between their web services to ensure they are available and the application services are running properly.

- Front-End - Authentication Service.
- Front- End – VCS
- Authentication Service - VCS
- Authentication Service - HSM
- VCS – RCG
- VCS – Ballot Box database
- VCS – HSM
- RCG – RCG database
- RCG – HSM
- RCG – SMS gateway
- RCG – DIFI web service

Regarding these processes:

- They can be executed individually for each server and component.
- Processes 1 and 2 (functionality and integrity, and cryptographic keys) shall be executed only once an election is configured. Otherwise, checks will be failing.
  The third process (connectivity) is independent of the election configuration, so it can be executed at any time.
- It is not necessary a special user-permission for executing these tests. The applications are open to be verified.
- The results of these processes are shown on screen, and registered at the application immutable logs.

6. Domain isolation

eVoting components are isolating the critical functions in two different ways, physical and logical isolation.

6.1 Physical Isolation.

There are three physical isolated areas for the eVoting components:

1) **e-Voting Collection Domain** components are deployed at Brønnøysund service provider. This is the main area for the voting services used by the voter.
2) The systems for the **e-Voting Receipt Generation Domain** (EVRGD) are deployed at DSB service provider. This service shall be isolated from the e-Voting Collection Domain.
3) Tallying Servers from the **e-Voting isolated Domain** (EVID) which are in KRD datacenter in Oslo.

6.2 Logical Isolation.

The system networks will be isolated to ensure voters are not able to access to internal services, and segment most critical eVoting components in different networks. By this way:

1) **Voters are only able to connect to the Front-End**, which will be managing the voters requests to the Authentication Service and the Vote Collector Server.
2) **Return Code Generators will not be connected to the Public Network Domain**, and they are only connected to the eVoting Collection Domain through a specific VPN to the Vote Collector Servers.
3) **Each one of the components from the e-Voting isolated Domain** (cleansing, mixing, and counting) shall be isolated from any other components from any domain. These components will only sharing information with other components through external devices (usb pendrives or external hard-drives).

7. Self-protection

The authenticity and integrity of the eVoting software which is running on eVoting servers or in the voter's PC, is something critical for the eVoting Security.

To protect the software integrity, there are three kinds of technical measures in place:

1. Configuration protection.
2. Software protection.
3. Segregation of duties protection.

7.1 Configuration protection.

Election configuration files will be digitally signed by the Administration Board. This signature will be verified by a service-check functionality which is executed at all the online components when the components are starting. It can be also reviewed on-demand according to an auditor or election operator requirements.

In addition to this, since some of the configurations from the Election Configuration Files are stored at a database, this service-check also reviews that the content of the database is the same than the digitally signed election configuration file.

7.2 Software protection.

1. The software applications running on eVoting Servers will be installed from **digitally signed RPM packages**, so it is not possible to modify these packages and not be detected.
   These secure installation packages will apply to:
   - Authentication Service
   - Vote Collector Server
   - Return Code Generator
   - Cleansing Service
   - Mixing Service
   - Counting Service

2. Once the software has been installed at each component, an Advanced Intrusion Detection Environment "**AIDE**" system will be implemented to monitor any change to the critical eVoting servers files. This system creates a baseline dictionary with a hash of the server files, and review this hashes in a periodical manner.

3. All the servers of the eVoting will have **TPM** chips (Trusted Platform Modules). These chips are containing a cryptographic key-pair which will be used to digitally sign and the AIDE reports. This AIDE reports will be periodically sent to a central TPM console, which will review these reports and document all the results in the immutable logs – which are sent to the central auditing environment.

All these mechanisms are ensuring the integrity and authenticity of the software application files.

7.3 Segregation of duties protection.

To ensure all components are having the expected behavior, and ensure otherwise it would be detected - in case they would be tampered or corrupted by non-TSF code or entities - the eVoting protocol has been designed to ensure enough segregation of duties interaction in a way that each component is reviewing the operations done by other component.

By this way, the e-voting process is distributed and reviewed through following components:

- The Authentication Server.
    - Reviews the MinID or poll-site authentication token.

- The VCS.
    - Verifies the Authentication token generated by the Authentication Token, and the MinID or poll-site authentication token.
    - Verifies the vote.
    - Creates zero knowledge proofs from the vote, and digitally sign these proofs.
    - Verifies the digital signature of the voting receipt generated by the RCG.

- The RCG
    - Performs the same vote’s and authentication token’s verifications than the VCS.
    - Verifies the zero knowledge proofs and the digital signature from the VCS.
    - Creates a digitally signed voting receipt.

- The voting applet.
    - Verifies the digital signature of the voting receipt generated by the RCG.

8. Non-bypassibility

To ensure the non-bypassibility ot the eVoting security, we have considered security controls at each infrastructure component with their own security objectives:

- **Voting Client – Applet**: The voting client is directly controlling all voter activities, so it will ensure the voter is fulfilling the election rules (e.g. regarding contests and number of selections). The vote is encrypted and digitally signed in the voting client.
  In addition to this, the applet will confirm the voter the result of the vote sending (successful or failed) and will verify that the voting receipt is properly signed by the RCG and it is related to the voting options selected by the voter.

- **Authentication Service**: The authentication service verifies the authentication token provided by MinID service or by the poll-site officers, and checks the voter eligibility against the Electoral Roll service published by the Admin system.

- **Vote Collector Server**: The VCS shall ensure all votes are valid votes before they are stored in the Ballot Box, by reviewing its signatures, voter certificates, authentication tokens, electoral roll… In addition, it creates and digitally signs some cryptographic proofs from the vote which are sent to the RCG, and reviews the digital signature of the voting receipt generated by the RCG.

- **Return Code Generator**: The RCG performs the same vote's validations than the VCS, and validates the cryptographic proofs generated by the VCS. In addition, it creates and stores a voting receipt which will be validated by the VCS and the voting Client, and obtains the Return Codes which will be sent to the voter to allow the voter verifiability.

- **Cleansing**: The cleansing performs a double function:
  - Verify all the votes to ensure they are valid votes, by reviewing its signatures, voter certificates, authentication tokens, electoral roll…
  - Select only one vote per voter and contest, by selecting the preferential vote per each voter (last e-vote, or an e-vote cast from a poll-site, or anyone in case the voter has cast a paper vote). A cleansed Ballot Box is created with these selected votes, without any information which could link the voter with her vote (digital signature, timestamp…)

- **Mixing**: The mixing service verifies the integrity and authenticity of the cleansed ballot box, and generates some cryptographic proofs while performing the mixing operations to allow an auditor verify the process.

- **Counting**: Verifies the integrity and authenticity of the cleansed ballot box, and reconstructs the election private key with the Electoral Board shares to decrypt the votes. While the votes are being decrypted, the counting generates some cryptographic proofs to allow an auditor verify the process.
  During the counting of decrypted votes, they are also verified to ensure the election rules are fulfilled (e.g. regarding contests and number of selections).

- **General Infrastructure**: All the infrastructure components which are supporting the eVoting software are hardened to avoid any security attacks which could be affecting software or data integrity.

The list of the security controls implemented in eVoting to ensure non-bypassibility is the following:

Applet	
A1	The Applet at the client-side **checks data inputs** from user to ensure it has not been manipulated (e.g. max number of selected options …).
A2	The Applet at the client side **verifies the text input** from the voter, against security attacks (malicious code injection).
A4	The applet request for **vote casting confirmation**, in order to avoid / prevent voter mistakes.
A5	The votes are **encrypted** at the client-applet with the Election Public Key.
A6	The votes are **digitally signed** at the client-applet, with the voter private key.
A7	Encrypted vote is linked with the voter information through the **schnorr signature** (which is verified with the zero knowledge proof).
A8	If the Applet does not receive a Voting Confirmation (considering a timeout period) the voter is notified.
A10	The voting applet verifies the digital signature of the voting receipt.
Authentication Server	
AS1	The Authentication Server requests a valid MinID Assertion signed by MinID service.
AS2	The MinID token is validated at the authentication server against a token-used list (token is single-use).
AS3	The Authentication Server requests to the MinID service the SAML assertion related to the MinID token, and verifies the SSN and the "audience" of the MinID token to ensure the token has been requested by the Authentication Service.
AS4	The Authentication Server checks the Electoral Roll at user-authentication process, to ensure the voter is authorized to vote.
AS5	The digital signature of the Electoral Roll Server is verified by the Authentication Server.
AS6	The voter contest information is included at the authentication token signed by the authentication service.
AS7	The Authentication Service checks that the MinID timestamp is not expired.
AS8	The Authentication Service request to MinID to force voters authentication on each connection - without applying single sign-on features.
AS9	Critical Operations are registered into the log files.
AS10	The application functionality for starting and ending the polling phase is performed automatically according to the Election config dates and times.
VCS	
VCS1	The authentication token and MinID token are validated at the VCS against a token-used list (token is single-use).
VCS2	The Vote Collector Server verifies the voter digital certificate with the voter digital certificates sotred at VCS
VCS3	The VCS checks the received vote parameters (integrity, signature, timestamp, MinID token, authentication token, ...).
VCS4	The Vote Collector Service checks that the Authentication Token and the MinID token are linked to the voter referred at the vote digital signature.
VCS5	The VCS checks the digital signature of the vote, so it cannot be modified and not detected.
VCS6	Vote Schnorr signature verification (zero knowledge proof)
VCS7	Votes are stored encrypted in the Ballot Box, and only the Electoral Board can decrypt them.
VCS8	The vote storage is registered at the immutable logs with a hash value of the vote content.

VCS9	The VCS does not store the ballot until the reception of the voting receipt from the RCG.
VCS10	The Voting Confirmation is not sent to the voter until the vote has been successfully stored.
VCS11	VCS will perform cross-checkings at the service startup, including a integrity verification of the configuration file vs configuration database, and the verification of the configuration file digital signature.
VCS12	The integrity of the ballot box is verified periodically.
VCS13	Critical Operations are registered into the log files.
VCS14	Unfinished operations are rolled back in case of Ballot Box malfunction or failure
VCS15	Unfinished operations are rolled back in case of VCS malfunction or failure
VCS16	Unfinished operations are rolled back in case of local Electoral Roll malfunction or failure
VCS17	Access to import Configuration files at VCS will be restricted, through a role-based mechanism.
VCS18	The Election Configuration file shall be digitally signed by authorized electoral board. These signatures are checked at VCS when the configuration is imported and applied.
VCS19	The XML files containing the voting options (to be processed by the client applet), are signed by the Vote Collector Server.
VCS20	The application functionality for ending the polling phase is performed automatically, according to the election date and time configuration.
VCS21	The application does not allow to export the ballot box until the polling phase has been ended.
VCS22	The ballot box is digitally signed by VCS and then it is exported.
VCS23	Access to application functionality for exporting the ballot box is restricted.
VCS24	The application functionality for starting and ending the polling phase is performed automatically according to the Election config dates and times.
VCS25	The VCS verifies all the information and signatures from the Authentication token and the MinID token
VCS26	The VCS verifies all timestamps related to the vote and the authentication and MinID tokens.
VCS27	The VCS verifies the voter is casting a vote for a contest which is autorized to cast a vote.
VCS28	The voter session ends after a certain inactivity period. After this time, any voting activity would be rejected and the voter would be redirected to authenticate again.
Key Mgmt Service	
KMS1	The Election Private Key is protected through cryptographic shared keys.
KMS2	The Election public key included into the applet, election configurations, and published components, will be signed by the Key Mgmt Service.
KMS3	The key management service will be a temporary service with restricted access (for key generation and distribution processes).
KMS4	Cryptographic keys are digitally signed by the A.Board at the Key Management Service.
KMS5	Key Mgmt Service - The private keys will be removed from the Key management service machine once they have been generated.
KMS6	The cryptographic keys generated in the KMS are distributed to the components in an encrypted P12 format, through the Key Stores.
Cleansing	
CL1	The ballot box is digitally signed by the VCS, and this signature is verified at the Cleansing Process.
CL2	The digital signature of the Electoral Roll is verified at the cleansing process.
CL3	The voting receipts are digitally signed by the RCG, and this signature is verified at the Cleansing Process.

CL4	The cleansing process verifies the votes digital signature by the voters.
CL5	The digital signature of the paper-Electoral Roll is verified at the cleansing process.
CL6	The cleansing process checks the Electoral Roll to ensure the voter is authorized to vote at the contest he/she has voted.
CL7	The cleansing process verifies the timestamp from the vote.
CL8	The cleansing process verifies authentication token and the MinID token. The cleansing process checks that the voter digital signature is linked to the corresponding authentication token.
CL9	The cleansing process verifies that the votes have a corresponding Voting Receipt stored in the RCG.
CL11	The cleansing processes will generate information (report) regarding the votes contained at the ballot box, and the results of the cleansing process, per voting groups (locations).
CL12	The cleansed ballot box shall be digitally signed by the Cleansing Process. This signature shall not be performed if the cleansing process is not complete.
CL13	The application does not allow to export the ballot box until the cleansing process has been ended.
MIXING	
MX1	The Mixing Process verifies the digital signature of the cleansed ballot box.
MX2	The mixing process performs self-testing of the mixing results.
MX3	The information from voters and voting options are shuffled at the Mixing Service; in that way, the mixed information which is going to be decrypted, will not contain private information.
MX4	The application functionality for validate the mixing operations is restricted to the auditor.
MX5	The mixed ballot box is digitally signed by the mixing service, just after the mixing process.
MX6	The mixing process will generate information regarding voting groups (locations) and number of votes processed per group.
MX7	Critical Operations are registered into the log files.
MX8	The application does not allow to export the ballot box until the mixing process has been ended.
MX9	The access to upload a cleansed ballot box in the mixing process is restricted through a RBAC token.
RCG	
RCG1	The RCG checks the received vote parameters (integrity, signature, timestamp, MinID token, authentication token, ...).
RCG2	The authentication token and MinID token are validated at the RCG against a token-used list (token is single-use).
RCG3	The Return Code Generator checks that the Authentication Token and the MinID token are linked to the voter referred at the vote digital signature.
RCG4	The RCG checks the digital signature of the vote, so it cannot be modified and not detected.
RCG5	Vote Schnorr signature verification (zero knowledge proof)
RCG6	The RCG generates a Voting receipt which will be sent to the voter and verified at the cleansing process.
RCG7	RCG verifies that the same vote has not been received previously
RCG8	Stored Voting Receipts are digitally signed by RCG.
RCG9	Voting options are confirmed through return codes (individualized for each voter)
RCG10	Vote casting is confirmed through an alternative channel (SMS).
RCG11	Return Codes are stored encrypted with a value which is only generated when the vote is cast.
RCG12	RCG verifies the return codes format before the return codes are sent.

RCG13	Return Codes are signed by the RCG.
RCG14	Voting cards are deleted when generated.
RCG15	Access to application functionality for exporting the voting receipts is restricted.
RCG16	The application does not allow to export the voting receipts until the polling phase has been ended.
RCG17	The voting receipt list is digitally signed when exported.
RCG18	Unfinished operations are rolled back in case of RCG malfunction or failure
RCG19	Critical Operations are registered into the log files.
RCG20	Access to import Configuration files at RCG will be restricted
RCG21	Integrity verification of the configuration file - vs. configuration information on the database.
RCG22	The digital signature of the Election Configuration file is checked at RCG
RCG23	The RCG verifies the Authentication token and the MinID token
RCG24	The RCG verifies all timestamps related to the vote and the authentication and MinID tokens.
RCG25	The RCG verifies the voter is casting a vote for a contest which is autorized to cast a vote.
RCG26	The request of the voters' phone numbers by RCG is encrypted and digitally signed.
RCG27	Voters' phone numbers are obtained from the DIFI webservice - encrypted and digitally signed.
Counting	
CO1	The counting service verifies the digital signature of the mixed ballot box.
CO2	The counting service verifies that the vote information is correct and has not been manipulated (number of candidates, candidates' codes). Otherwise, non-valid votes are rejected.
CO3	Only the Electoral Board members are able to decrypt the ballot box.
CO4	Counting Critical Operations are registered into the log files.
CO5	The counting process verifies the digital signature of the decrypted ballot box.
CO6	The counting process is digitally signing the vote counting.
CO7	The administration Board is digitally signing the decrypted ballot box at the counting service.
Controlled Environment	
IP1	At the controlled environment, the information (identification token) included in the smart cards is digitally signed by the polling stations.
INFRASTUCTURE	
INFR1	The integrity of critical files from all eVoting servers is monitored through AIDE software
INFR2	AIDE reports from all eVoting servers are digitally signed by keys stored in the TPM, and reported to central TPM consoles
INFR3	External Connections from the Datacenter to voters and external services are protected with SSL
INFR4	The operating systems of the eVoting servers are properly hardened to restrict the access to authorized personnel only
INFR5	The operating systems of the eVoting servers are properly hardened to register adequate logs of any important activity
INFR7	The databases of the eVoting servers are properly hardened to restrict the access to authorized personnel only
INFR6	The databases of the eVoting servers are properly hardened to register adequate logs of any important activity

INFR10	All the eVoting servers are synchronized to a reliable NTP server
INFR8	A protected and restricted VPN connection is the only communication between VCS and RCG servers.
INFR9	Most critical files are monitored through iNotify functionality ensuring quick notification if they are manipulated
INFR11	Only the Authentication Server has access to the voters key-pairs (P12 files)
INFR12	The eVoting components are deployed in high-availability mode, where all the resources are duplicated in different datacenters, according to load-estimations, and implementing load-balancers
INFR13	Load-balacers, firewalls, and web servers shall be hardened to reject denial of service attacks
INFR14	Most critical infrastructure components are isolated from any network and operated in a controlled environment
INFR15	Most critical cryptographic keys used by servers are stored in Hardware Security Modules (HSM)
INFR16	Private keys of the Electoral board members are stored in smartcards protected by PIN selected by them