6

Requirements for Remote Electronic Voting

The previous chapter specifies the requirements for stand-alone direct recording electronic voting machines. Based on this list and as a result of the development procedures from Sect. 4.1 this chapter provides the requirements for remote electronic voting systems. This standardised, consistent, and exhaustive list of requirements respects the glossary and syntax introduced in Sect. C and 4.4 respectively. The partition of this chapter is equal to the one from chapter 5: Before providing the list of requirements, the chapter-specific notation is explained (meaning those notations not used in the requirements for stand-alone direct recording electronic voting machines). Then, the exact target of evaluation is defined. The partition for the requirements is also taken over from the previous section: First, the security and functional requirements are defined and then, the assurance, usability, and operational requirements.

6.1 Citation and Additional Notations

Phase 3 in the requirement development process for remote electronic voting systems (the first draft of requirements), is based on the requirements for stand-alone direct recording electronic voting machines from chapter 5[1]. Additionally, the requirements listed in the GI/BSI/DFKI Protection Profile [161][2] are considered as part of phase 4 of the development process (improvement based on existing literature).

Notation. As stated in Sect. 4.4, only those requirements from existing catalogues which do not already appear in chapter 5 (because they only address

[1] Note, two requirements categorised as security requirements in Chap. 5 are shifted to functional ones in this chapter, namely: O.T.AvailInfo and O.T.SepDuty. Vise versa, O.OSP.DeleteRecord is shifted from the list of functional requirements to the list of security requirements.

[2] The GI/BSI/DFKI Protection Profile is introduced and discussed in Sect. 8.2.

M. Volkamer: Evaluation of Electronic Voting, LNBIP 30, pp. 93–113, 2009.

remote electronic voting) are referred in this chapter. To indicate the relationship between the requirements from the GI/BSI/DFKI Protection Profile [161] the security and functional requirements are labelled with "BSI name". Correspondingly, "Chap5 name" refers to requirements in the previous chapter for stand-alone direct recording electronic voting machines.

The security and functional requirements are distinguished between those involving the polling phase and those involving 'only' the phase after the polling. Moreover, the requirements are further distinguished according to the component that is addressed: the remote electronic voting system in general, the voting server, the tallying software the client-side voting software, or the audit system.

6.2 Target of Evaluation

The electronic voting system focused on in this chapter is called remote electronic voting system as defined in Sect. 2.1. The idea is to use such a system in parallel to postal voting, that is, every voter who is eligible to cast a postal vote can now choose between the postal or the electronic channel.

A remote electronic voting system can provide more or less functionality. Systems addressed here do not cover all possible implementation techniques and not all election phases. This section describes the target of evaluation.

Covered Functionality. The following operations for the poll workers are addressed:

- Identification and authentication
- Starting the polling phase
- Making a selection on the ballot
- Resuming the polling phase after any kind of exceptions, malfunction, or breakdown
- Checking the system state
- Closing the polling phase
- Starting the tallying phase

In addition, the following operations for the voters are addressed:

- Identification and authentication
- Changing a selection before casting
- Inducing vote casting
- Casting the vote
- Cancelling the voting process

Functionality Not Covered. The election setup and archiving phase are not addressed in the later security and functional requirements definition (analogously to the target of evaluation description in Sect. 5.2). Additionally, the following functionalities which might be implemented in some remote electronic voting systems are out of the scope of the following examinations:

- Running two or more polls in parallel[3]
- Keeping confidential who cast a vote
- Voter or universal verification procedures
- Resistant against disputations
- Changing the electoral register during the polling phase
- Statistical data collection
- Vote updating

If these are implemented, the requirement set needs to be adapted. For instance, to enable the application of more than one poll in parallel demands at least the following additional requirement: the ·*remote electronic voting system*· `shall` prevent anyone from linking different ·*e-votes*· from the same ·*voter*· to one another when polls are run. This is caused by the following threat: an inside intruder with access to ·*e-votes*· after the ·*polling phase*· discovers some aspect of ·*voters'*· identities by examining ·*votes*· that were cast together. For instance, non-citizen residents may have limited voting rights. An intruder could determine which votes came from a particular community.

Covered Techniques. Section 2.4 illustrate that there is no best solution for the voter authentication, to ensure the secrecy of the vote, and to implement the client-side voting software. Thus, it is tried to allow all of these techniques for the target of evaluation. Any of the authorisation techniques from Sect. 2.4, are possible implementations. The supported techniques to ensure the secrecy of the vote according to Sect. 2.4 are 'anonymisation in the polling phase and the tallying phase'. The possible voting client techniques, according to Sect. 2.4, are the 'thin and the fat client' approach: that is, computations on the client-side are required.

Techniques Not Covered. Systems implementing "anonymisation in the election setup phase" as a technique to ensure the secrecy of the vote are not covered. For those systems, some of the requirements can be removed because they are passed already by design decisions. However, corresponding requirements need to be defined for the election setup phase. The application of Web browser solutions is only possible if some of the requirements are defined as assumptions about the environment. This is further discussed in Sect. 8.2.3.

Scope. According to the description in Sect. 2.4, a remote electronic voting system includes the voting server (hardware and software), the client-side voting software, the vote-casting device, and the tallying software.

'One' Voting Server. This section only considers one voting server to be generic and to match as many different remote electronic voting systems as possible. However, existing remote electronic voting systems usually distinguish between two or even more voting servers: some are generic and provide n voting servers depending on the configuration. Note, in the case of more

[3] Therefore, the security requirements O.T.LinkInParalElec from Sect. 5.3.2 is not further discussed for remote electronic voting systems.

than one voting server where the voting servers communicate with each other, additional requirements for this communication must be added.

Assumptions. Remote electronic voting belongs to the voting forms where the voter casts his vote in an unprotected environment. As it is proposed to apply remote electronic voting only in parallel to postal voting, problems and corresponding requirements (like coercion resistance) caused by unprotected environments are not addressed in the requirement definition as these problems are already accepted within postal voting (assumption A.ProtectedEnvironment). In addition, for the further considerations, it is assumed that if the remote electronic voting system is set up correctly, it contains the proper electoral register and candidate list as well as the proper definition of valid and invalid votes (assumptions A.ProperConfig). However, there is a requirement (O.OSP.SelfCheck) demanding that poll workers have the possibility of checking the configuration before starting the polling phase. Moreover, it is assumed that (if necessary) the distribution of identification and authentication tokens succeeded and, thus, only but all voters have an identification and authentication token (assumption A.AuthToken).

6.3 Security Requirements

6.3.1 Security Requirements for the Polling Phase

(a) Security Requirements for the Remote Electronic Voting System

T.IneligVoter: An ·*ineligible voter*· ·*casts*· a ·*vote*· in order to affect the ·*election result*·.	BSI T.UnauthorisedVoter
O.T.IneligVoter [eq] The ·*remote electronic voting system*· `shall` unambiguously identify and authenticate the ·*voter*· before storing his ·*vote*· in the ·*e-ballot box*·.	CoE [82, 94a] PTB VP [1-1] BSI O.UnauthorisedVoter
T.OneVoterOneVote: A malicious ·*elector*· ·*casts*· a second ·*vote*· in order to affect the ·*election result*·.	BSI T.UnauthorisedVoter Chap5 T.UnauthVotesA
O.T.OneVoterOneVote [eq] The ·*remote electronic voting system*· `shall` store in the ·*e-ballot box*· only one ·*vote*· per ·*voter*·; it `shall` store the first received ·*vote*· per ·*voter*·.	CoE [5b] BSI O.UnauthorisedVoter Chap5 O.T.UnauthVotes
T.UnauthVotes: An inside intruder adds ·*e-votes*· to the ·*e-ballot box*· at the ·*voting server*· in order to affect the ·*election result*·.	BSI A.ElectionOfficers Chap5 T.UnauthVotesB

O.T.UnauthVotes [di] The ·*remote electronic voting system*· `shall` store in the ·*e-ballot box*· only ·*e-votes*· cast from ·*eligible voters*·. Any other access to the ·*e-ballot box*· `shall` be denied.	Chap5 O.T.UnauthVotes
T.PersonalDataNet: An outside intruder sniffs the network in order to collect personal data from ·*voters*·.	–
O.T.PersonalDataNet [dp] The ·*remote electronic voting system*· `shall` ensure the data protection law with respect to the transmission of any personal data.	BSI O.SecretMessage
T.SecretAuthNet: An outside intruder sniffs the network to get ·*authentication information*· and to use this to ·*cast*· a ·*vote*· on behalf of a ·*voter*· in order to affect the ·*election result*·.	–
O.T.SecretAuthNet [un] The ·*remote electronic voting system*· `shall` protect the confidentiality of the transmitted ·*authentication information*·.	BSI O.SecretMessage
T.IntResultNet: The outside intruder sniffs the network in order to compute intermediate results.	BSI T.SecretMessage
O.T.IntResultNet [fr] The ·*remote electronic voting system*· `shall` ensure the confidentiality of the transmitted ·*e-votes*· during the ·*polling phase*·.	BSI O.SecretMessage
T.DeleteMsgNet: Unnoticed, the outside intruder deletes messages in the network to exclude ·*voters*· from the ·*election*· in order to affect the ·*election result*· or in order to confuse ·*voters*·.	BSI T.IntegrityMessage
O.T.DeleteMsgNet [un] [tr] The ·*remote electronic voting system*· `shall` ensure that protocol messages cannot be deleted undetected.	PTB VP[4-6a], DR[1-2b] BSI T.IntegrityMessage
T.AlterMsgNet: An outside intruder unnoticed replays old protocol messages, sends new ones, or alters messages in order to affect the ·*election result*·.	BSI T.IntegrityMessage

O.T.AlterMsgNet [all] The ·*remote electronic voting system*· `shall` verify the freshness, authenticity, integrity, and format correctness of all messages before processing them.	BSI O.IntegrityMessage
T.DeleteRecord: An outside intruder uses the ·*voter's*· ·*vote-casting device*· after the ·*voter*· ·*cast*· his ·*vote*· in order to compromise the secrecy of the vote.	BSI A.Buffer
O.T.DeleteRecord [se] The ·*remote electronic voting system*· `shall` delete any records related to the ·*voter's*· ·*voting process*· from the ·*vote-casting device*· when finishing the ·*voting process*·.	Chap5 O.OSP.DeleteRecord
T.ElecSecrecyNet: An outside intruder sniffs the network in order to compromise the secrecy of the vote.	BSI T.SecretMessage
O.T.ElecSecrecyNet [fr] The ·*remote electronic voting system*· `shall` not provide any information in the transmitted protocol messages, which allows to construct the link between a particular ·*voter*· and his ·*vote*·. The ·*remote electronic voting system*· `shall` ensure that neither the ·*vote*· itself nor the number of chosen ·*voting options*· (including an empty ·*ballot*·), nor a ·*spoilt*· ·*vote*· (for example, by using the length of the protocol messages) can be linked to a particular ·*voter*·. In addition, it `shall` be ensured that the sequence of messages does not reveal the link.	BSI O.SecrecyOfVoting
T.ProofGenA: A malicious ·*elector*· uses all information either sent to, displayed on, and/or sent from his ·*vote-casting device*· to construct a proof in order to sell his ·*vote*·.	BSI T.Proof
T.ProofGenB: A malicious ·*elector*· uses all information from T.ProofGenA and intermediate results calculated on his ·*vote-casting device*· to construct a proof in order to sell his ·*vote*·.	BSI T.Proof
O.T.ProofGen [se] The ·*remote electronic voting system*· `shall` ensure that ·*voters*· are not able to construct a receipt proving their ·*vote*·. Neither information sent to, displayed on, sent from, nor intermediate results calculated on his ·*vote-casting device*· or protocol messages sequences `shall` serve as proof.	CoE [93b] BSI O.Proof Chap5 O.OSP.DeleteRecord

(b) Security Requirements for the Voting Server

T.WrongSW: An outside intruder disseminates manipulated ·*client-side voting software*· in order to reach any of his goals.	–
O.T.WrongSW [all] The ·*voting server*· `shall` communicate only with the authentic and unaltered ·*client-side voting software*·.	–
T.TamperServerA: An outside intruder gets access to the ·*voting server*· over the network and tampers with it in arbitrary ways in order to reach any of his goals.	BSI A.ElectionServer
T.TamperServerB: An inside intruder tampers with the ·*voting server*· in arbitrary ways in order to reach any of his goals.	BSI A.ElecttionOfficers Chap5 T.Tamper
O.T.TamperServer [all] The ·*voting server*· `should` be tamper-resistant. The ·*voting server*· `shall` be tamper-evident.	PTB VP[2-3] Chap5 O.T.Tamper
T.AC: An outside intruder gets access to the ·*voting server*· without knowing or having the access tokens to tamper with the ·*voting server*· in order to reach any of his goals.	BSI P.AuthElectionOfficers A.ServerRoom Chap5 T.AC
O.T.AC [all] The ·*voting server*· `shall` implement an access control policy for the ·*poll worker interface*· which • restricts all activities to particular ·*user*·-roles and • requires physical presence.	BSI O.AuthElectionOfficers Chap5 O.T.AC
T.ElectionSecrecyA: An outside intruder accesses the ·*election data*· after the ·*polling phase*· in order to compromise the secrecy of the vote.	BSI T.ArchivingSecrecyOfV. Chap5 T.ElectionSecrecy
T.ElectionSecrecyB: An inside intruder gets access to the ·*voting server*· and uses stored information in order to compromise the secrecy of the vote.	BSI A.ElectionOfficers Chap5 T.ElectionSecrecy

O.T.ElectionSecrecy [se] The ·*voting server*· `should` not store any information which could link the ·*voter*· with his ·*vote*· after the completion of the ·*voting process*·. Where any information which could link the ·*voter*· to his ·*vote*· is stored on the ·*voting server*·, it `shall` only be accessible to those with appropriate authority.	BSI O.ArchivingSecrecyOfV. Chap5 O.T.ElectionSecrecy

(c) Security Requirements on the Client-Side

T.TamperClient: An outside intruder runs malware on the ·*vote-casting device*·, which either reads the ·*vote*· (in order to compromise the secrecy of the vote), alters the ·*vote*·, or reads the authentication information to ·*cast*· a ·*vote*· or to bar the ·*voter*· from ·*casting a vote*· (in order to affect the ·*election result*·).	BSI A.VoteCastingDevice Chap5 T.Tamper
O.T.TamperClient [all] The ·*client-side voting software*· `shall` ensure that its operations and data are unaffected by other applications running on the ·*vote-casting device*·.	Chap5 O.T.Tamper
T.WrongServer: An outside intruder tries to redirect the ·*voter*· to a faked ·*voting server*· in order to reach any of his goals.	BSI T.AuthenticityServer
O.T.WrongServer [all] The ·*client-side voting software*· `shall` only communicate with the authentic and unaltered ·*voting server*·.	CoE [90b] BSI O.AuthenticityServer

6.3.2 Security Requirements for the Tallying Phase

T.IntegVotes: An inside intruder tampers with ·*e-votes*· after the ·*polling phase*· and before the ·*tallying phase*· in order to affect the ·*election result*·.	Chap5 T.IntegVotes
O.T.IntegVotes [di] The ·*voting server*· `shall` protect the integrity and authenticity of ·*e-votes*· after the ·*polling phase*·.	BSI O.ArchivingIntegrity Chap5 O.T.IntegVotes
O.T.AuthCheckCount [di] The ·*tallying software*· `shall` verify the integrity and authenticity of ·*e-votes*·.	Chap5 O.T.AuthCheckC.

T.IntegElecData: An inside intruder tampers with ·*election data*· after the ·*tallying phase*· in order to affect the ·*election result*· in case of recounts.

BSI T.ArchivingIntegrity
Chap5 T.IntegElecData

O.T.IntegElecData [di] The ·*tallying software*· shall protect the integrity and authenticity of ·*election data*· as soon as the tallying is completed.

BSI O.ArchivingIntegrity
Chap5 O.T.IntegElecData

T.AffectCounting: An inside intruder installs malware on the machine running the ·*tallying software*· in order to affect the ·*election result*·.

BSI A.ElectionOfficers
Chap5 T.AffectCounting

O.T.AffectCounting [di] The ·*tallying software*· shall ensure that its operations and data are unaffected by other applications.

Chap5 O.T.AffectCounting

6.4 Functional Requirements

6.4.1 Functional Requirements for the Polling Phase

(a) Functional Requirements for the Remote Electronic Voting System

O.OSP.VoteRight [un] [di] The ·*remote electronic voting system*· shall ensure that no ·*voter*· looses his voting right without having ·*cast a vote*·.

BSI P/O.OneVoterOneVote

O.OSP.NoInteract [un] The ·*remote electronic voting system*· shall prevent ·*voter*· interactions in case of exceptions and malfunctions.

Chap5 O.OSP.NoInteract

O.OSP.Confirmation [tr] The ·*remote electronic voting system*· shall provide a confirmation to the ·*voter*· regarding the status of his ·*vote*· – at least the information that his ·*e-vote*· has been successfully stored.

Appl. Note: In case the ·*voter*· does not receive the confirmation, he shall get this information as soon as he logs on again.

BSI P/O.Acknowledgement
Chap5 O.OSP.PosFeedback

O.OSP.Feedback [un] The ·*remote electronic voting system*· shall provide feedback to the ·*poll workers*· in form of error messages in case of exceptions, malfunctions, and breakdowns. Where a ·*voter*· is in the ·*voting process*· at that time he shall also get a feedback.

Chap5 O.OSP.NegFeedback

O.OSP.DataLoss [di] The ·*remote electronic voting system*· shall prevent data loss during normal operations and in case of exceptions, malfunctions, and breakdowns.

PTB VP [2-1b]
BSI A/OE.DataStorage
Chap5 O.OSP.DataLoss

O.OSP.Availability [un] [non-core] The ·*remote electronic voting system*· should be available during the whole ·*polling phase*·.

Appl. Note: The ·*remote electronic voting system*· shall be robust against power outage at the ·*voting server*·, unexpected ·*user*· activity, environmental effects (for instance, mechanical, electromagnetic, and climatic) to the ·*voting server*·, and network problems.

PTB CF [2-1]
BSI A/OE.Availability
Chap5 O.OSP.Avaliability
Chap5 O.OSP.Robust

O.OSP.VoteRightExc [un] [di] The ·*remote electronic voting system*· shall ensure that in case of exceptions, malfunctions, and breakdowns no ·*voter*· looses his right to ·*cast*· a ·*vote*· nor get the possibility to ·*cast*· two ·*votes*·.

Appl. Note: The ·*remote electronic voting system*· shall be capable to determine whether a particular ·*voter*· ·*cast*· a vote and his ·*e-vote*· was successfully stored in case of exceptions, malfunctions, and breakdowns.

PTB VP[1-5]
BSI P/O.OneVoterOneVote
Chap5 O.OSP.LastVote

(b) Functional Requirements for the Voting Server

O.OSP.SepDuty [all] The access control mechanism shall only allow access to the ·*voting server*· if at least two different ·*users*· are logged on.

BSI P/O.AuthElectionO.
Chap5 O.T.SepDuty

O.OSP.Auditing [tr] The ·*voting server*· shall be capable of producing comprehensive audit data.

BSI P/O.Audit
Chap5 O.OSP.Autditng

O.OSP.InfoPW [di] [non-core] The ·*voting server*· shall indicate to the ·*poll worker*·

- the number of ·*votes*· ·*cast*· so far and
- its current state.

Chap5 O.OSP.InfoPW

O.OSP.StoreAllVotes [di] The·*voting server*· shall store in the ·*e-ballot box*· all ·*e-votes*· ·*cast*· by ·*eligible voters*· during the ·*polling phase*·.

Chap5 O.OSP.StoreAllVotes

O.OSP.PWClosePoll [un] The ·*poll worker interface*· shall warn the ·*poll workers*· if they try to close the ·*election*· before the final date.

BSI P/O.EndingElection
Chap5 O.OSP.PWClosePoll

O.OSP.AvailInfo [tr] The ·*voting server*· shall not provide any information about the ·*voting process*· except the current state and the number of ·*votes*· ·*cast*· so far.

BSI P/O.SecrecyOfVotingElec.
Chap5 O.T.AvailInfo

O.OSP.SelfCheck [all] The ·*voting server*· should regularly perform automatic self-checks and report the results to the ·*poll workers*·. The ·*voting server*· shall be capable of performing self-checks.

BSI P/O.Failure
Chap5 O.OSP.SelfCheck

O.OSP.ErrorRecovery [di] [un] The ·*voting server*· shall run a self-check before a resuming is possible. In case of irreversible problems the ·*voting server*· shall prevent a resuming of the ·*polling phase*·.

BSI P.Failure
Chap5 O.OSP.ErrorRecovery

O.OSP.PWInterface [se] [fr] The only functionality provided by the ·*poll worker interface*· is

- identification and authentication,
- starting the ·*polling phase*· which is only possible once,
- resuming the ·*polling phase*· after any kind of exceptions, malfunctions, and breakdowns according to O.OSP.ErrorRecovery,
- closing the ·*polling phase*· after which the actions ‘starting’ and ‘resuming’ are disabled,
- starting the ·*tallying phase*· only after having closed the ·*polling phase*·,
- performing self-checks,
- checking that the ·*voting server*· has been set up correctly (for example, order of ·*voting options*· and empty ·*e-ballot box*·),
- checking the current state according to O.OSP.InfoPW, and
- reading the audit trails.

Appl. Note: The ·*voting server*· shall not provide any functionality to reach any of the intruder’s goals described in Sect. 4.3.

BSI P/O.EndOfElection
P/O.IntegrityElectionOfficers
P/O.IntermediateResult
P/O.Failure
P/O.Audit
P/O.StartTallying
Chap5 O.OSP.PWInterface
O.OSP.PWCheck

O.OSP.SecrecyAfterBreakd [se] In case of exceptions, malfunctions, and breakdowns, the ·*voting server*· `shall` not reveal the link from the last ·*voter*· to his ·*selections*· or ·*vote*·.

Chap5 O.T.SecrecyAfterBr.

O.OSP.ClosePoll [un] [non-core] The acceptance of ·*e-votes*· into the ·*e-ballot box*· `should` remain open for a sufficient phase of time to allow for any delay of data transport.

PTB DR[1-2a]
CoE [96b]

O.OSP.AdequNoVotes [un] [non-core] The ·*voting server*· `shall` be capable of recording an adequate number of ·*votes*·.

Chap5 O.OSP.AdequNoVotes

O.OSP.AdequNoBallotOpt [fr] [non-core] The ·*voting server*· `shall` support an adequate number of ·*voting options*·.

Chap5 O.OSP.AdequNoBall.

(c) Functional Requirements for the Client-Side Voting Software

O.OSP.Interface [fr] The ·*client-side voting software*· `shall` provide the following functionality for the ·*voter*·:

- Identification and authentication
- Make a choice on the ·*ballot*·
- Change ·*selections*· before ·*casting a vote*·
- Initialise vote casting
- ·*Vote casting*·
- Cancel his ·*voting process*· at any time

BSI P/O.Abort,
P/O.Correction,
P/O.OverhasteProtection
Chap5 O.OSP.V-Interface

O.OSP.AccurDisp [fr] The ·*voting server*· `shall` accurately display the authentic and unaltered ·*ballot*·.

Chap5 O.OSP.AccurDisp

O.OSP.Transmission [un] The ·*client-side voting software*· `shall` immediately transmit the ·*e-votes*· to the ·*voting server*·, whenever a ·*voter*· has ·*cast*· his ·*vote*·.

–

O.OSP.Spoil [fr] [non-core] The ·*client-side voting software*· `should` provide the functionality for the ·*voter*· to ·*spoil*· his ·*vote*·.

Chap5 O.OSP.Spoil

O.OSP.SpoilWarning [fr] [non-core] The ·*client-side voting software*· `should` warn the ·*voter*· when he tries to ·*spoil*· his ·*vote*· in one or more ·*polls*·.

Chap5 O.OSP.SpoilWarning

O.OSP.EqualPres [fr] The ·*client-side voting software*· `shall` ensure equality and accuracy of presentation of ·*voting options*· on any ·*vote-casting device*·.

Appl. Note: The ·*remote electronic voting system*· shall avoid the display of other influencing messages.

Chap5 O.OSP.EqualPres

O.OSP.AccurRep [fr] The ·*client-side voting software*· `shall` ensure that the ·*voter's*· ·*selections*· are accurately represented in the ·*e-vote*·.

Chap5 O.OSP.AccurRep

O.OSP.CompatClient [fr] [non-core] The ·*client-side voting software*· `should` be compatible with any ·*vote-casting device*· and with devices used by people with disabilities where appropriate.

Chap5 O.OSP.CompatClient

6.4.2 Functional Requirements for the Tallying Phase

O.OSP.ReadToOtherSystems [tr] The ·*remote electronic voting system*· `shall` provide the functionality to upload ·*e-votes*· into any ·*tallying software*·.

Chap5 O.OSP.ReadToO.

O.OSP.DeleteData [di] The ·*voting server*· `shall` provide the functionality to completely delete all data from previous ·*elections*·.

Chap5 O.OSP.Delete

O.OSP.AccurCalc [di] The ·*tallying software*· `shall` accurately calculate results using the appropriate algorithm based on all (authorised) ·*e-votes*· stored in the ·*e-ballot box*· and only based on these ·*e-votes*·.

BSI P/O.Tallying
Chap5 O.OSP.AccurCalc

6.4.3 Functional Requirements for the Audit System

O.OSP.Audit1 [tr] The ·*audit system*· `shall` provide the functionality to record, monitor, and verify audit data.

Chap5 OSP.Audit.1

O.OSP.Audit2 [tr] The ·*audit system*· `shall` protect the integrity and authenticity of audit records.

Chap5 O.OSP.Audit.2

O.OSP.Audit3 [tr] The ·*audit system*· `shall` have access to a reliable time source.

Chap5 O.OSP.Audit.3
BSI OE.SystemTime

O.OSP.Audit4 [tr] The ·*audit system*· `shall` record system configuration (including software version numbers) and ·*election*· configuration (including ·*voting option*· information) on the ·*voting server*· at least at the following points

- beginning and end of ·*polling phase*·, as well as
- before and after tallying.

Chap5 O.OSP.Audit.4

O.OSP.Audit5 [tr] The ·*audit system*· `shall` check the ·*e-ballot box*·, the ·*ballot*· content, and the ·*authentication data*· for evidence of tampering.

Chap5 O.OSP.Audit.5

O.OSP.Audit6 [tr] The ·*audit system*· and its records `should` be tamper-resistant and `shall` be tamper-evident.

Chap5 O.OSP.Audit.6
BSI OE.AuditTrailProt.

O.OSP.Audit7 [tr] For every action performed by ·*poll workers*· the ·*audit system*· `shall` record

- a timestamp,
- the nature of the action, and
- the ID of the particular ·*poll worker*·(where available).

Chap5 O.OSP.Audit.7

O.OSP.Audit8 [tr] The ·*audit system*· `shall` record (with timestamps, where appropriate)

- breakdowns,
- exceptions,
- malfunctions, and
- results of any self-checks.

Chap5 O.OSP.Audit.8

O.OSP.Audit9 [tr] The ·*audit system*· `shall` implement the access control policy defined by the ·*responsible election authority*·.

Chap5 O.OSP.Audit.9

O.OSP.Audit10 [tr] The ·*audit system*· `should` not record any information which might endanger the secrecy of the vote. Where such information is stored it `shall` only be accessible to those with appropriate authority.

Chap5 O.OSP.Audit.10

O.OSP.Audit11 [dp] The ·*audit system*· `shall` ensure the data protection law.

CoE [110]

6.5 Assurance Requirements

Some of the assurance requirements are additionally labelled with small letters. These are used in Sect. 7.3 to refer to parts of a particular requirement.

Assur.1 [all] The ·*responsible election authority*· `shall` define the trust model for their particular ·*election*·.

Chap5 Assur.1

Assur.2 [un] The ·*manufacturer*· `shall` develop the ·*electronic voting system*· (a) according to software engineering best practice, including use of (b) version control, and (c) bug tracking for all documents and source code.

Chap5 Assur.2

Assur.3 [all] The ·*manufacturer*· `shall` produce the following documents ensuring that they are exhaustive, consistent, unambiguous, appropriate, comprehensible, and concise: Chap5 Assur.3

(a) Complete system specification
(b) Implemented security functions
(c) Requirement conformance claim
(d) Description of each component
(e) Environmental assumptions
(f) Testing record
(g) Development security measures
(h) User-guide containing
- normal use instructions for all ·*users*· for all phases
- appropriate responses to all system messages

(i) delivery procedure

Assur.4 [un] The ·*manufacturer*· `shall` build the ·*electronic voting system*· from reliable components. Chap5 Assur.4

Assur.5 [tr] The ·*manufacturer*· `shall` disclose (a) the documentation from Assur.2, (b) executable program, (c) source code, (d) bug tracking, and (e) version control (at least to the ·*testing authority*·). Chap5 Assur.5

Assur.6 [all] The ·*manufacturer*· `shall` test the ·*electronic voting system*·, including functional and usability tests. Chap5 Assur.6

Assur.7 [fr] [un] [non-core] The ·*manufacturer*· `should` involve ·*users*· in the interface development process. Chap5 Assur.7

Assur.8 [all] The ·*testing authority*· `shall` do a risk analysis based on the threat model. Chap5 Assur.8

Assur.9 [all] The ·*manufacturer*· `shall` limit the functionality of the ·*electronic voting system*· and ·*tallying software*· to that necessary for the ·*election*·.

Chap5 Assur.9

Assur.10 [all] The ·*testing authority*· `shall` evaluate the ·*electronic voting machines*· against the requirements. Tests `shall` include penetration, and usability tests.

Chap5 Assur.10

Assur.11 [all] The ·*testing authority*· `shall` examine the ·*manufacturer's*· (a) documentation from Assur.2, (b) executable program, (c) source code, (d) bug tracking, and (e) version control for compliance with requirements and software engineering best practice.

Chap5 Assur.11

Assur.12 [all] The ·*testing authority*· `shall` examine (a) the delivery procedures for the ·*electronic voting system*·, (b) the identified development security measures, and (c) the applied software engineering approach.

Chap5 Assur.12

6.6 Additional Requirements

6.6.1 Usability Requirements

Usab.1 [un] All user interfaces `shall` be user-friendly.

Chap5 Usab.1

Usab.2 [un] [fr] All system messages provided by all user interfaces `shall` be understandable.

Chap5 Usab.2

Usab.3 [un] The ·*vote-casting interface*· `shall` make provision for ·*voters*· with disabilities.

Chap5 Usab.3

Usab.4 [tr] The ·*vote-casting interface*· `shall` provide immediate feedback to the ·*voter*· regarding the status of his ·*vote*·.

Chap5 Usab.5

Usab.5 [fr] The ·*vote-casting interface*· `shall` protect the ·*voter*· from accidentally ·*casting*· his ·*vote*· Chap5 Usab.6

Usab.6 [all] The ·*poll worker interface*· `shall` protect the ·*poll workers*· from taking any action accidentally. Chap5 Usab.7

Usab.7 [un] [tr] All used methods `shall` be efficient, thus, the ·*voting process*· does not take more time as necessary. Chap5 Usab.8

Usab.8 [un] The ·*client-side voting software*· `shall` be easy to install on the ·*vote-casting device*·. Chap5 Usab.9

6.6.2 Operational Requirements

Op.1 [tr] The ·*responsible election authority*· `shall` develop a contingency plan describing appropriate responses to at least the following circumstances: Chap5 Op.5

- results produced by recount or alternative ·*tallying software*· do not agree with original result
- number of ·*votes*· recorded does not match number of ·*electors*·
- any kind of exceptions, malfunctions, and breakdowns

Op.2 [all] The ·*responsible election authority*· `shall` define (for all ·*election*· phases): Chap5 Op.7

- timetables
- access control policy (including separation of duties and minimum team size) inclusive audit data and system related access control
- administration activities
- ·*user*· roles
- key management policy
- incident levels
- reporting procedures

Op.3 [un] The ·*responsible election authority*· `shall` provide additional channels to ·*cast*· the ·*vote*· other than the remote electronic voting one.

CoE [4]

Op.4 [all] The ·*responsible election authority*· `shall` develop procedures covering all stages of the ·*election*·, including

- secure ·*voting server*· storage at all times
- ·*voting server*· configuration (including ·*ballot*· details, order on ·*voting server*·, and ·*tallying software*·)
- checking ·*voting server*· (including configuration and empty ·*e-ballot box*·)
- response to any kind of exceptions, malfunctions, and breakdowns
- recording of ·*poll worker*· activities, ·*voting server*· state changes, system resuming, etc.
- ensuring that the ·*voting server*· is in the appropriate state at every stage in the ·*election phase*·.
- closing the ·*poll(s)*·, including disabling ·*voting server*·
- tallying and re-tallying
- comparing number of ·*votes*· recorded with number of ·*electors*·
- ·*archiving phase*·, including data deletion at the end
- ·*identification and authentication token*· delivery, their storage and management where necessary

Chap5 Op.4

Op.5 [all] The ·*responsible election authority*· `shall` define all ·*responsible election authority*· variables, prescribe the certification process (including decertification and recertification), appoint the ·*testing authority*·, and the ·*certification authority*·.

Chap5 Op.6

Op.6 [un] The ·*responsible election authority*· `shall` coordinate the different channels, for instance, it `shall` prevent ·*voters*· ·*casting one vote*· per possible channel and `shall` develop a procedure to merge the results from different channels.

CoE [6, 7, 37, 41, 44, 45, 53]
PTB CF[2-4]

Op.7 [tr] [non-core] Before the ·*election*· the ·*responsible election authority*· `shall` publicly disclose all technical information about the ·*electronic voting system*· (including design, configuration, version numbers, etc.).
Remark: Exceptions are only acceptable where it can be shown that such a disclosure would either endanger the security of the ·*electronic voting system*· or genuinely endanger the intellectual property of the ·*manufacturer*·. Chap5 Op.8

Op.8 [all] The ·*responsible election authority*· `shall` educate ·*poll workers*· in the use of the ·*electronic voting system*· and `shall` ensure that information provided to them is understandable. Chap5 Op.1

Op.9 [di] The ·*responsible election authority*· `shall` ensure that ·*election data*· is stored with its authentication codes (and, where applicable, from the ·*tallying software*·) for the prescribed ·*archiving phase*·. Chap5 Op.2

Op.10 [all] The ·*poll workers*· `shall` follow the procedures described by the ·*responsible election authority*·. Chap5 Op.13

Op.11 [all] The ·*poll workers*· `shall` respond to system messages in accordance with the user-guide. Chap5 Op.14

Op.12 [fr] [un] The ·*responsible election authority*· `shall` educate ·*voters*· in the use of the ·*electronic voting system*· and `shall` ensure that the information provided to them is understandable. Chap5 Op.3

Op.13 [tr] [non-core] The ·*responsible election authority*· `should` arrange alternative ·*tallying software*· to check results. Chap5 Op.9

Op.14 [un] [non-core] The ·*responsible election authority*· `shall` clearly indicate whether the ·*electronic voting system*· are being used in a real ·*election*·. Chap5 Op.10

Op.15 [fr] [non-core] The ·*responsible election authority*· `should` ensure that all ·*electronic voting system*· display the ·*ballot*· in a uniform way. Chap5 Op.11

6.7 Summary

This chapter defines the exact type of considered remote electronic voting systems and itemises all requirements for this type of electronic voting systems. This list contains 71 system requirements (while these are divided in 21 security requirements, 42 functional requirements, and eight usability requirements), 12 assurance and 15 operational requirements. According to Chap. 4 the requirements are labelled by election principle(s). The requirements refer either direct or indirect to corresponding requirements in in [37], [143], and [62] (indirect by referring to requirements from Chap. 5).

Section 6.1 clarifies the relationship between the requirements in this chapter and those provided in the GI/BSI/DFKI Protection Profile [161]. The notations used to refer to the Protection Profile and to requirements from the previous chapter are introduced. Afterwards, Sect. 6.2 describes the exact target of evaluation: the considered functionality for voters and poll workers is defined and it is stated that systems implementing "anonymisation in the election setup phase" as a technique to ensure the secrecy of the vote are not covered as well as systems using the Web browser approach, while all other approaches discussed in Chap. 2.4 are considered. Moreover, it is decided that the evaluation only covers the functionality for the polling phase and the tallying phase. Besides these functional aspects, it is explained why different possible voting servers are subsumed to one voting server. In addition, the assumptions to the environment are presented (A.ProtectedEnvironment, A.ProperConfig, and A.AuthToken).

The 21 security requirements in Sect. 6.3 are deduced from corresponding threats which are also specified. These requirements are divided into those for the polling phase and those for the tallying phase. The functional requirements in Sect. 6.4 are composed of 28 requirements for the polling phase, three requirements for the tallying phase, and 11 requirements for the audit system. Assurance requirements in Sect. 6.5 address either the tasks of the manufacturer (and thus the development process), the testing authority (how to evaluate the system), or the responsible election authority. In addition, Sect. 6.6 specifies the list of usability and organisational requirements. The last category addresses only responsible election authority tasks and mainly document and procedures to define.

As the focus of this book is on security issues, the security, functional, and assurance requirements are treated as input for the next part – the evaluation part – while the organisational and usability requirements are not further discussed.