Examination of the Swiss Internet Voting System

Audit scope 2a (development process)

Follow-up audit (round 2)

02.11.2022 / v1.0FINAL

Work performed for:

Swiss Federal Chancellery
Political Rights Section
Federal Palace West Wing
3003 Bern

Contact information

SCRT SA
Rue du Sablon 4
1110 Morges
Switzerland

T: +41 21 802 64 01
E: info@scrt.ch

Authors and contributors

Antonio Fontes	Lead examiner

Management summary

Scope and objective of the examination

During the period August 2021 - March 2022, SCRT carried out a security audit of the Swiss Post's e-voting system against a subset of requirements set forth in the Federal Chancellery's ordinance on e-voting (scope 2a: software development processes).

Six (6) non-compliances (findings) were identified and reported, along with a set of eleven (11) recommendations.

The objective of this second examination was to follow-up on the findings and recommendations raised during the first-round examination.

Methodology

The examiners looked for evidence of effort to comply with the criteria that were not yet fulfilled during the initial examination. They performed live interviews, both on-site and virtual, and reviewed the relating documentation.

The examination was performed during September and October 2022.

Due to the negligible impact of the revised ordinance that came into force in July 2022 on this scope in particular, the conclusions of this audit are already compatible with these changes.

Results

After the second round of audit, four (4) recommendations were found implemented, five (5) recommendations were found partially implemented or in the process of being soon implemented, and two (2) recommendations were found pending implementation, leaving four (4) ordinance requirements assessed as either missing or partially missing.

Non-compliances were observed for the following requirements:

- Requirement 24.1.1 (secure development lifecycle): A subset of controls commonly considered as essential in the secure systems development lifecycle are missing or incompletely implemented (i.e., establish a reference threat model, integrate threat modelling, establish a security vetting process for third-party components).
- Requirement 24.1.20 (security documentation): The security whitepaper for the e-voting system is still being produced.
- Requirement 24.5 (quality assurance): Processes to confirm the good operation of security attestation tools used to test the e-voting system are missing.
- Requirement 17.2 (test coverage): Processes to assess new design proposals and tools to perform runtime security testing are still being evaluated and formalized.

All reported non-compliances are classified as *moderate* severity.

Final note

The examiners emphasize on the fact that a large portion of the recommendations issued during the first round of examination can typically take several months, if not more than a year, to be implemented satisfyingly. Some of the recommendations required the adoption of tools often considered as organization-wide purchases, which undergo complex vetting and selection processes. Other recommendations required the training of internal resources into new techniques and adapting their existing processes.

For these reasons, the examiners anticipated that a significant part of the recommendations would be *work in progress*.

Still, assuming the recommendations will be implemented as announced and within reasonable delay, the examiners consider the outcome of the examination to be generally positive.

The examiners conclude this summary by thanking Swiss Post again, and more particularly those who have been personally involved in this follow-up audit, for their cooperation and for the transparency demonstrated throughout the examination.

Table of content

1 Context

1. Electronic voting (hereafter referred to as: e-voting) was introduced in Switzerland through multiple pilot schemes from 2004 onwards. A total of 15 cantons made e-voting possible in over 300 trials, until early 2019. Two implementations were available: the system made by the canton of Geneva and the system operated by the Swiss Post (initially developed by Scytl). In June 2019, the canton of Geneva announced the withdrawal of its e-voting system with immediate effect. It was followed in July of the same year by the announcement by Swiss Post of the withdrawal of its e-voting system from operation to focus on improving the solution. Since then, e-voting is not possible in Switzerland.

2. In June 2019, the Swiss Federal Chancellery (hereafter: Federal Chancellery) was commissioned by the Federal Council to redesign a new trial phase, using “e-voting systems, which are fully verifiable” [1]. This redesign of the trial phase focuses on four objectives:

   1. Further development of the e-voting systems
   2. Effective controls and monitoring
   3. Increased transparency and trust
   4. Stronger connection with the scientific community

3. A taskforce was set up to make proposals for the future of internet voting. To that end, the Federal Chancellery invited experts from academia and industry to engage in a broad dialogue on internet voting in Switzerland. After this dialog, the Federal Chancellery and the cantons published a final report on the redesign and relaunch of internet voting trials, with a catalogue of measures [2].

4. The Federal Council took note of the final report and commissioned the Federal Chancellery to amend the legal bases of the Confederation. In April 2021, the Federal Council opened a consultation procedure on the amendment to the legal bases, which was drafted by the Federal Chancellery. The redesign includes both a partial revision of the Ordinance on Political Rights (PoRo) [3] and a complete revision of the Federal Chancellery Ordinance on Electronic Voting (“VEleS”, or “OEV”) [4]. The OEV specifies, among others, the requirements for authorising electronic voting, including the technical and administrative controls for approving an e-voting system[1].

5. The Federal Chancellery issued an audit concept for the examination of Swiss internet voting systems [5] defining the foundations for assessing the compliance of electronic voting systems with the draft OEV and its annex [6], as per chapter 26 of the annex of the draft OEV, and for obtaining recommendations for improvements.

[1] The criteria for the examination reported in this document is built on a subset of these controls.

6. SCRT was mandated by the Federal Chancellery to assess the compliance of the Swiss Post's revamped e-voting system against some of the requirements of the draft OEV. The present report focusses on the examination of the perimeter defined as follows in the audit concept: *Scope 2a - Development process*.

7. In May 2022, the Federal Council enacted the partially revised Ordinance on Political Rights (PoRo) [7], which became applicable from Jul. 1st 2022. The totally revised Federal Chancellery Ordinance on Electronic Voting (OEV) [8] came into force on the same date.

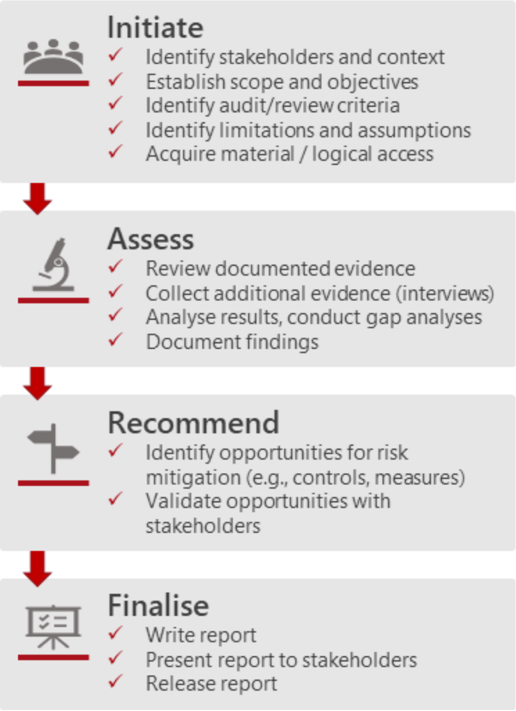
8. A second assessment was conducted in mid-September 2022 to follow-up on the findings raised in the initial audit report [9].

2 Methodology

9. The methodology to assess the requirements remains unchanged from the first-round examination. Readers of the round 1 examination report [9] can skip this section.

2.1 General process

10. The examination was based on SCRT's information systems audit methodology. The process specifies four-phases depicted in the figure below:



Initiate
- ✓ Identify stakeholders and context
- ✓ Establish scope and objectives
- ✓ Identify audit/review criteria
- ✓ Identify limitations and assumptions
- ✓ Acquire material / logical access

Assess
- ✓ Review documented evidence
- ✓ Collect additional evidence (interviews)
- ✓ Analyse results, conduct gap analyses
- ✓ Document findings

Recommend
- ✓ Identify opportunities for risk mitigation (e.g., controls, measures)
- ✓ Validate opportunities with stakeholders

Finalise
- ✓ Write report
- ✓ Present report to stakeholders
- ✓ Release report

Figure 1 - Process

2.2 Collection of evidence

11. As a general principle, the examiners aimed at acquiring two types of evidence for each requirement. Types of evidence included documents (e.g., documentation, test reports, written instructions, etc.), statements (e.g., obtained during plenary sessions or during interviews), and demonstrations (e.g., tools, scripts, configurations or process material shown during interviews).

12. Part of the examination included reviewing documents classified as confidential by Swiss Post and thus not released to the public. Motives for not disclosing these documents to the public included either or both the a) preservation of the confidentiality of business processes deployed at the organisation level and which may confer Swiss Post a competitive advantage

on other actors, and b) the preservation of confidentiality of operational data (e.g., risk control, infrastructure operations, etc.). Swiss Post confirmed to the examiners that these documents remain accessible to the Cantons.

13. Unless specified otherwise, written evidence collected and reviewed during the examination is referred in the bibliography of this report, with public links whenever possible. Some sources, which were not made physically available to the examiners but shown on-screen during live interviews with the examinees, are cited without reference.

14. Swiss Post provided the examiners an internal document mapping each Federal Chancellery requirement with one or more corresponding documented evidences [10].

2.3 Requirements

15. The examiners specified three states for the requirements, as follows:

- » *Pass* - The proposed evidence allows the examiners to consider that the requirement is met.
- » *Partial miss or P.Miss* – The proposed evidence allows the examiners to conclude that the requirement is partially met.
- » *Miss* - The proposed evidence either allows the examiners to conclude that the requirement is not met (explicit miss), or that there is insufficient evidence to reach another conclusion (implicit miss).

2.4 Findings

Definition

16. During the first-round examination, the examiners raised a finding when evidence provided by the examinee did not provide satisfying assurance that the requirement was met (implicit miss) or when the evidence provided explicitly indicated that the requirement was not met, or partially met (explicit miss).

17. New findings were not identified during the second-round examination. Any reference to findings in this document refers to findings already reported in the first-round report.

Severity of findings

18. The examiners specified three severity levels, as follows:

- » *High severity* - The finding identifies a failure to produce evidence of satisfying a requirement.
- » *Moderate severity* - The finding identifies a partial failure to produce evidence of satisfying a requirement.

- *Low severity* - The finding identifies a notable opportunity for improvement or optimisation.

19. Readers should note that the severity indicated in this report only reflects the opinion of the examiners and could be subject to re-evaluation by relevant parties.

Recommendations

20. For each finding, one or more recommendations were issued during the first-round examination. A finding is considered as resolved or closed once all its associated recommendations are implemented.

2.5 Assumptions

Trustworthiness of statements

21. The examiners assume that the examinees were completely honest and transparent when providing answers to the examiners' assessment questions. Although several proofs of testing were shown, no observation of the actual implementation of the OEV's requirements within the e-voting system was carried out to verify the accuracy of the examinees' statements.

Trustworthiness of security measures

22. The examiners assume that the security measures described in the documents provided as evidence in the context of the present examination are implemented and are effective. No verification of the actual implementation effectiveness of the OEV's requirements within the e-voting system (e.g., security testing, vulnerability assessment, penetration test, etc.) was carried out within the scope of this examination to verify the accuracy of the statements made in the security documents.

Non-regression

23. The examiners assumed that the Federal Chancellery requirements, for which satisfying evidence of implementation was provided by Swiss Post during the first round of examination, were still met during the second round of examination.

3 Examination criteria

24. The audit concept for the e-voting examination [11] specifies several assessment scopes (e.g., cryptographic protocol, software, infrastructure, etc.) , each encompassing a subset of the requirements specified in the Federal Chancellery ordinance on electronic voting published in April 2022 [8].

25. This chapter enumerates the requirements for the scope 2a ("assess the development processes").

3.1 Effect of the revised ordinance on initial results (impact assessment)

26. A revised ordinance on electronic voting entered into force on 1st July 2022 [8]. Changes between the revised version and the version used during the first-round examination are summarized as follows[2]:

- Requirement 8.12 (transparent communication of known flaws): renamed to 8.13.
- Requirement 24.1.19 (configuration list): requires an additional information item labelled as "commit history".

27. The examiners assessed the impact of the changes above and concluded as follows:

- Regarding requirement 8.13: impact is considered null.
- Regarding requirement 24.1.19: the configuration management system used by the examinee to produce the artifacts can produce a history of commits to the source code (including the source code of the CI/CD pipeline itself), along with their author, as part of its feature set. Impact is also considered null.

3.2 Second round examination criteria

28. As no significant changes were made to the criteria for the second round of examination, and assuming there were no regressions (see: 2.5 – Assumptions), the examiners limited the criteria for the second examination to the requirements that had not been met, or partially met, during the first-round examination (findings and associated recommendations from the first-round examination report). The illustration below describes this situation:

[2] This list shows changes to the requirements that are part of the assessment scope 2a. Changes to the ordinance that affected other examination scopes are not listed here.

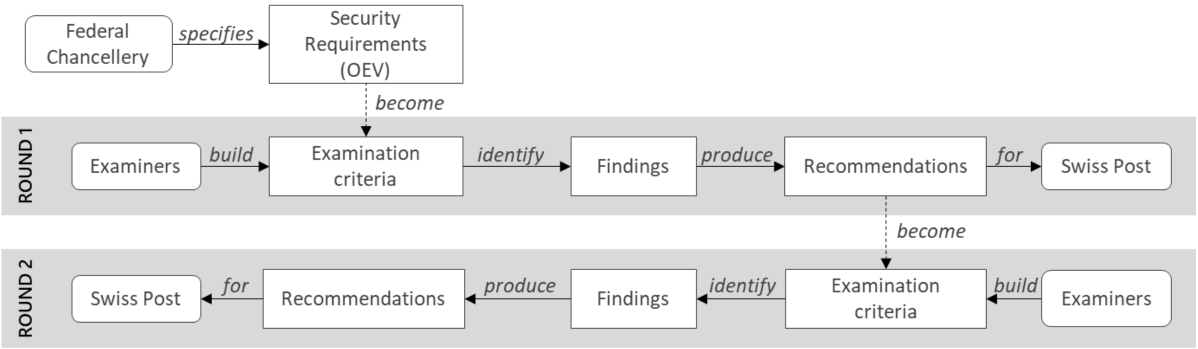


Figure 2 – Examination criteria during first and second rounds of examination

3.3 Federal Chancellery requirements

29. The following tables enumerate the 17 requirements that constitute the criteria for scope 2a ("assess the development process") of the audit concept for the second phase of the examination of the Swiss internet voting systems [11], grouped by topic[3].

Development process and lifecycle requirements

Key	Requirement
24.1.1	A life cycle model is defined. The life cycle model: » is used for the development and maintenance of the software (a); » provides for the necessary controls during the development and maintenance of the software (b); » is documented (c).
24.1.2	A list must be made of the development tools used and configuration options chosen for the use of each development tool.
24.1.3	The documentation for the development tools includes: » a definition of the development tool (a); » a description of all conventions and directives used in the implementation of the development tool (b); » a clear description of the significance of all configuration options for using the development tool (c).
24.1.4	The implementation standards to be applied must be specified.

Table 1 - E-voting requirements: lifecycle

[3] Topic names and the regrouping of requirements were chosen by the examiners and may not necessarily reflect the Federal Chancellery's vision.

Software security documentation requirements

Key	Requirement
24.1.20	Software development security documentation includes: » a description of the physical, procedural, personnel, and other security measures necessary to protect and ensure the integrity of the design and implementation of the software in its development environment (a); » evidence that the security measures provide the necessary level of protection to preserve the integrity of the software (b).

Table 2 - E-voting requirements: software security documentation

Quality assurance requirements

Key	Requirement
24.5	Regular and objective checks are carried out to ensure that the processes carried out and the associated work products comply with the description of the processes, standards and procedures to be implemented (a). Deviations are followed up until they are corrected (b).

Table 3 - E-voting requirements: quality assurance

Configuration management system requirements

Key	Requirement
24.1.14	The software is provided with a unique identification.
24.1.15	The configuration management documentation includes: » a description of how configuration items are identified (a); » a configuration management plan describing how the configuration management system will be used in the development of the software and the procedures that will be followed for the adoption of changes or new elements (b); » evidence that the procedures for adoption provide for adequate review of changes for all configuration items (c).
24.1.16	The configuration management system: » uniquely identifies all configuration items (a); » provides automated measures to ensure that only authorised changes are made to configuration items (b); » supports the development of the software through automated procedures (c); » ensures that the person responsible for accepting the configuration item is not the same person who developed it (d); » identifies the configuration items that make up the security functions (e); » supports verification of all changes to the software using automated procedures, including logging of the author and the date and time of the change (f); » provides an automated method for identifying any configuration items that are affected by a change to a particular configuration item (g);

	» can identify the version of the source code on the basis of which the software is generated (h).
24.1.17	All configuration items are inventoried in the configuration management system.
24.1.18	The configuration management system is used in accordance with the configuration management plan.
24.1.19	A configuration list is created that contains the following items: » the software, » evidence of the checks required to ensure security compliance, » the parts that make up the software, » the source code, » the commit history[4], » reports on security flaws and on the status of their correction (a). For each element relevant to security functions, the developer is named (b). Each element is uniquely identified (c).

Table 4 - E-voting requirements: configuration management system

Testing requirements

Key	Requirement
17.1	The functions relevant to the security of the system (security functions) are tested. The tests are documented with test plans and expected and actual test results. (a) The test plan (b): » specifies the tests to be performed; » describes the scenarios for each test, including any dependencies on the results of other tests. The expected results must show the results that are expected if the test is successfully executed. (c) The actual results must be consistent with expected results. (d)
17.2	An analysis must be made of the test coverage. This includes evidence that: » the tests defined in the test documentation match the functional specifications of the interfaces (a); » all interfaces have been fully tested (b).
17.3	An analysis must be made of the depth of testing. This includes evidence that: » the tests defined in the test documentation match the subsystems related to security functions and modules that play a role in ensuring security (a); » all subsystems related to the security functions mentioned in the specifications have been tested (b); » all modules that play a role in ensuring security have been tested (c).
25.13.3	The integration tests cover all modules.

[4] The requirement for a commit history was added in the revised ordinance.

25.13.4	The software tests cover all modules.

Table 5 - E-voting requirements: testing

Transparency requirement

Key	Requirement
8.13[5]	Known flaws and the need for action associated with them are communicated transparently. **: identified as 8.12 in the round 1 examination criteria.*

Table 6 - E-voting requirements: transparency

Systematic correction of flaws requirements

Key	Requirement
24.4.1	Processes are defined for the correction of flaws. The processes include: » documentation of specific aspects, in particular with regard to the traceability of flaws for all versions of the software, and of the methods used to ensure that system users have information on flaws, corrections and possible corrective actions (a); » the obligation to describe the nature and impact of all security flaws, information on the status of work to find a solution and the corrective measures adopted (b); » a description of how system users can make reports and enquiries about suspected flaws in the software known to the software developers (c); » a procedure requiring a timely response and automatic dispatch of security flaw reports and appropriate corrective actions to registered system users who may be affected by the flaw (d).
24.4.2	A process is defined for handling reported flaws (a). This process ensures that all reported and confirmed flaws are corrected and that the procedures for correction are communicated to system users (b). It provides for arrangements to ensure that the correction of security flaws does not give rise to new security flaws (c).
24.4.3	Policies must be defined for the reporting and correction of flaws. These include: » instructions on how system users can report suspected security flaws to the developer (a); » instructions on how system users can register with the developer to receive reports of security flaws and the corrections (b); » details of specific contact points for all reports and inquiries on security issues concerning the software (c).

Table 7 - E-voting requirements: systematic correction of flaws

[5] Identified as requirement key 8.12 in the round 1 examination report, the requirement remains unchanged.

3.4 Additional criteria: secure systems development lifecycle

30. A central part of the work consisted in evaluating the development process put in place by Swiss Post for its e-voting system. To limit potentially ambiguous interpretations of what could qualify as the OEV requirement "life cycle model, which provides for the necessary controls during maintenance and development of the software" [6, p. 33], the examiners derived an interpretation of "necessary controls" based on the following references:

- Security software lifecycle requirements and assessment procedures (v1.0), PCI [12],
- Application software security controls, Critical security controls (v.8), CIS [13],
- Fundamental practices for secure software development (third edition), SAFECode [14].
- SAMM - Software assurance maturity model assessment tool (v1.5), OWASP [15].

31. The characterisation of "necessary controls" is summarised as follows:

Organisational measures:

- The organisation has appointed a security champion within each development team, which, among others, acts as a security liaison and leader between the development team and the organisation's security structures,
- The organisation established interfaces between the development team and the organisation's information security structures and with external security advisor(s),
- The organisation established interfaces with its incident response structure,
- The organisation ensures that personnel involved in the design, construction or attestation of the system attended role-based security awareness and training.

Operational enablement measures:

- A catalogue of threats, with their respective controls or countermeasures, and status (e.g., mitigated, not mitigated, etc.), is documented and maintained,
- Security requirements are documented,
- Secure design principles or secure architecture baseline requirements are documented and integrated in the development process,
- Changes to the system are subject to a threat assessment (e.g., threat modelling, abuse cases, attacker stories, etc.) aimed among others at identifying potential and relevant threats and identifying appropriate countermeasures,
- Coding guidelines, or equivalent, are documented. In particular, they propose standardised responses to well-known causes of risk and error (e.g., input canonicalization and validation, output encoding, command interpreter query parameterisation, filesystem access, database access, protected storage of sensitive data or secrets, etc.),
- High-risk code is identified as such and subject to extended review (e.g., manual review or testing),

- Vulnerability management is integrated and performed throughout the entire development process with the support of adequate tools and processes,
- Source code, including all relevant adjacent artifacts, released to customers is centralised, versioned, and protected from unauthorised access.

Attestation measures:

- Change requests are subject to standardised or routine security verification (e.g., security checklist in definition of ready).
- New code is tested for well-known vulnerabilities and errors (aka, source code review, static analysis, etc.) as well as existing code (to mitigate regressions) prior to release.
- Third-party components are vetted against well-known threats prior to being integrated into the system.
- Third-party components are inventoried and monitored for known issues or vulnerabilities.
- The runtime is tested for well-known vulnerabilities and errors (e.g., dynamic/runtime application security testing) prior to release.
- The security of the final system, both in its entirety and its individual high-risk components, is regularly tested by independent actors through adequate methods (e.g., external penetration testing, bug bounty, 3rd party expert review, etc.).
- Releases and all associated artifacts are certified, and their authenticity can be independently verified (e.g., code or binary signing, etc.),

3.5 Additional criteria: inclusion of third-party components

32. Due to the extensive use of third-party components in the e-voting system, the examiners also derived an interpretation of "necessary controls" for the use and inclusion of third-party components as part of the software engineering process. The following reference was used to derive requirements:

- Managing security risks inherent in the use of third-party component, SAFECode [16].

33. The characterisation of "necessary controls", in the context of third-party components, is summarised as follows:

Organisational measures:

- The organisation conducts a standardised risk assessment prior to integrating a third-party component into the system.

Operational enablement measures:

- Threats derived from the use and inclusion of third-party components in the software application are identified and documented, and adequate countermeasures or controls implemented wherever relevant and/or necessary.

Attestation measures:

- Third-party components are tested for known vulnerabilities or malicious activity, both prior to their inclusion in the system and after (monitoring).

4 Follow-up examination on March 2022 findings

34. This chapter summarizes the observations and conclusions made and reached by the examiners on the remediation of findings as part of the second-round examination. For each finding, the following content is included:

- The summary of the finding, as initially reported during the round 1 examination,
- The result of the follow-up examination on the finding and its recommendations, with details on the evidence provided to the examiners, the assessment details for each recommendation and its updated status of completion.

4.1 F-01 Insufficient integration of security in the software development lifecycle

Initial finding description (round 1 examination – March 2022)

Key	F-01
Title	Insufficient integration of security in the software development lifecycle
Requirement ID(s)	24.1.1
Requirement(s)	A life cycle model is defined. The life cycle model: » is used for the development and maintenance of the software (a); » **provides for the necessary controls during the development and maintenance of the software (b);** » is documented (c).
Severity	Moderate
Rationale	The examiners noted the accrued effort invested by the examinee to build a secure e-voting cryptographic protocol. However, the same level of effort was not observed equivalently on the overall system's development lifecycle of the solution [9, Para. 25]. In particular, the examiners noted the following: » the examinee invested a large amount of effort, controls and activities to secure the solution, and more particularly the e-voting protocol, but remained unable to position itself in terms of excellency and maturity on what is generally considered state-of-the-art in terms of secure systems engineering [9, Para. 25]; » the security testing strategy focused on post-code flaw/vulnerability detection measures (e.g., code review, public reviews, bug bounties,

	penetration testing by third parties, etc.) while [9, Para. 26], thus potentially failing to identify threats, flaws and vulnerabilities at earlier stages of the development lifecycle; » a lack of formal training, in particular amongst architects and developers of the solution [9, Para. 29];
Recommendations	R-01 - Formalise the integration of security development lifecycle guidance or best practices into the e-voting system's development process. R-02 - Ensure e-voting system personnel and stakeholders have received adequate role-based training on secure systems engineering. R-03 - Integrate threat modelling, or equivalent, in early stages of the development process. R-04 - Establish a baseline set of security principles or rules for each phase of the development lifecycle (e.g., requirements, architecture and/or design, coding, testing, build, deployment, etc.).

Table 8 - Finding F-01 (insufficient integration of security in the software development lifecycle)

Follow-up examination (round 2 – October 2022)

Evidence

35. The following evidence was provided or shown:

- Document: SAMM assessment project presentation [17]
- Document: SAMM Assessment results [18]
- Document: onboarding checklist for new employees [19]
- Document: Security whitepaper – Software development – e-voting [20]
- Document: Employee training records [21]
- Document: Secure design and coding guidelines [22]
- On-site interview

Assessment

36. On the first recommendation (R-01):

- The examinee benchmarked its systems engineering process for the development of the Swiss electronic voting using the OWASP SAMM methodology [23].
- In addition to acquire a formal visibility on its position amongst what appears to be considered by the community as the best practice in terms of secure systems engineering processes, the examinee identified and scheduled a set of actions to improve its maturity level.

37. Based on the evidence shown to the examiners, the implementation of recommendation R-01 is considered to be completed.

38. On the second recommendation (R-02):

- The examinee formalized a set of guidelines and instructions to address well-known security issues.
- E-voting project developers and architects are required to take part in regular awareness sessions held by the security champions.
- E-voting project members are now required to attend two application security trainings (general session + training to use the internal material). Their attendance is tracked and available through employee training records.
- Documentation with references to internal application security awareness material and documentation have been integrated into the new e-voting team member onboarding process.

39. Based on the evidence shown to the examiners, the implementation of recommendation R-02 is considered to be completed.

40. On the third recommendation (R-03):

- At the organization level, the examinee has started deploying threat modelling globally across its system engineering activity.
- The process is still in an early maturity stage (e.g., use of system-centric threat identification methods such as STRIDE[24]) and being tuned to better align with the organization’s constraints and culture (e.g., agile threat modelling activity).
- Within the e-voting project, threat modelling is not yet streamlined into the development process, but the practice is being adopted. The examiners could observe that security champions have been trained to apply threat modelling methods and that the organization is supporting this evolution.
- In terms of threat model, the examiners noted that the e-voting project still operates on the governance of the threat model specified by the Federal Chancellery in its e-voting ordinance [8, App. 13] and yet need to establish their own.

41. Based on the evidence and the proposed action plan shown to the examiners, the implementation of recommendation R-03 is considered to be in progress and aimed towards completion.

42. Regarding the fourth recommendation (R-04):

- The examinee has made available a set of internal rules [22] to help members of the e-voting project design and code software more securely. The material is available through the intranet.
- The examiners could confirm that the developers / architects are aware of the guidance and know how to reach it. Using the guidance is part of the training curriculum.
- The examiners could confirm that the material proposed to the e-voting team is under continuous improvement.

43. Based on the evidence shown to the examiners, the implementation of recommendation R-04 is considered to be completed.

4.2 F-02 Conflicting/ambiguous attribution of security responsibilities

Initial finding description (round 1 examination – March 2022)

Key	F-02
Title	Conflicting/ambiguous attribution of security responsibilities
Requirement ID(s)	24.1.1
Requirement(s)	A life cycle model is defined. The life cycle model: » is used for the development and maintenance of the software (a); » **provides for the necessary controls during the development and maintenance of the software (b);** » is documented (c).
Severity	Moderate
Rationale	The examiners noted security-related roles and responsibilities assigned to members of the e-voting project were both ambiguous and conflicting [9, Para. 36]. They were also diluted across multiple individual [9, Para. 37]. Additionally, the examiners noted that security champions were not clearly identified in the e-voting project team, and members relied on the support of an external systems engineering security expert, who was routinely involved as advisor to the project [9, Para. 38], [9, Para. 39].
Recommendations	R-05 - Establish a security champion program and appoint a champion in each e-voting system development team.

Table 9 - Finding F-02 (Conflicting/ambiguous attribution of security responsibilities)

Follow-up examination (round 2 – October 2022)

Evidence

44. The following evidence was provided:

- Document: Security whitepaper – Software development – e-voting [20]
- On-site interview

Assessment

45. On the recommendation (R-05):

- The examinee instated a security champions program across the organization and within the e-voting project.

- Three security champions are involved in the e-voting team: one, who is a member of the project and acts as the security champion of the project.
- A second security champion works at the organization level (*Enterprise Security Architecture*) and is involved at 20% in the e-voting project.
- The team still has access to the external expert/advisor already noted during the first-round examination [9, Para. 40].

46. Based on the evidence shown to the examiners, the implementation of recommendation R-05 is considered to be completed.

4.3 F-03 Insufficient protection measures against malicious third-party components

Initial finding description (round 1 examination – March 2022)

Key	F-03
Title	Insufficient protection measures against malicious third-party components
Requirement ID(s)	24.1.1[6]
Requirement(s)	A life cycle model is defined. The life cycle model: » is used for the development and maintenance of the software (a); » **provides for the necessary controls during the development and maintenance of the software (b);** » **is documented (c).**
Severity	Moderate
Rationale	The examiners noted the following: » The examinee deployed a software composition analysis (SCA) tool, which allowed the team to be alerted when a third-party component (TPC) is publicly identified as vulnerable or compromised [9, Para. 43]. » However, other classes of threats, such as TPCs that have been either compromised or built maliciously without knowledge of the community or the vendor, and TPCs that contain unreported vulnerabilities, were still lacking mitigating controls [9, Para. 43], [9, Para. 44]. » The examinee had not yet formalized the threat model on which decisions to adopt or reject TPCs in the e-voting platform were or would be based.

[6] A typo in the first-round examination report incorrectly mentioned requirement 24.1.15 (configuration management system documentation) in addition to requirement 24.1.1.

Recommendations	R-06 - Establish a reference threat model for the use of third-party components in the e-voting system, maintained with the status of implementation of chosen controls and countermeasures. R-07 - Establish a security vetting process for the selection of new third-party components, and the review of existing ones, embedded in the e-voting system.

Table 10 - Finding F-03 (Insufficient protection measures against malicious third-party components)

Follow-up examination (round 2 – October 2022)

Evidence

47. The following evidence was provided:

- Document: Security whitepaper – Software development – e-voting [20]
- On-site interview

Assessment

48. On the first recommendation (R-06):

- The examinee has not yet formalized its own reference threat model and uses the threat model proposed in the Federal Chancellery's ordinance [8, App. 13.19].
- The Federal Chancellery's threat model specifies one explicit threat scenario for software dependencies, as follows: "A backdoor is introduced into the system via a software dependency and is exploited by an external attacker to access the system." It was noted during the first examination round [9, Para. 44] that this scenario fails to include other threats which typically arise from the inclusion of third-party components in software deliverables.

49. Based on the evidence shown to the examiners, the implementation of recommendation R-06 is considered to be incomplete.

50. On the second recommendation (R-07):

- The examiners note that the examinee has initiated a process to reduce its use of TPCs used in the e-voting system. The current initiative aims reducing TPCs embedded in the deliverables by at least a third. The examinee is aware, however, that reducing the use of TPCs will not necessarily make the system more secure: where the exposure to third-party risk gets lower, the burden of writing and maintaining secure libraries gets higher. Still, this initiative increases Swiss Post's sovereignty over the content of its deliverables.

- The use of an SCA[7] tool to detect vulnerable dependencies during build operations was already noted during the first examination [9, Para. 70]. Swiss Post extended its implementation to also continuously monitor the system running in production. This measure helps reducing a delay often observed in organizations that deploy SCA only in their deployment pipeline, thus failing to detect their vulnerable state until a new build or deployment occurs.
- The insertion of a new library into the e-voting project is governed by the approval of architects at the organizational level. Developers and architects of the e-voting project are not given direct access to the community repository (e.g., Maven Central) but to an organization-wide repository acting as an intermediary authority.
- To be added to the organization's repository, a motivated request must be issued and submitted for validation by an architecture team in charge of the repository.
- The precise criteria through which architects approve or reject a component is not yet formalized. This led the examiners to believe that the fidelity of the validation process may vary. Although, a component that has previously been approved at the organization level (e.g., for another project) will not require additional specific authorization, which brings the risk that it may not have been vetted with the same risks in mind.
- During the interviews, assurance was given to the examiners that an initiative to specify the criteria through which the components are authorized had begun and the process would very likely be formalized early 2023.

51. Based on the evidence and the proposed action plan shown to the examiners, the implementation of recommendation R-07 is considered to be in progress and aimed towards completion.

[7] SCA, Software Composition Analysis: tooling used to gain visibility over third-party components (TPCs) embedded into a software application. In the context of the e-voting project, features such as component inventorying, vulnerability scanning, and alerting are particularly appreciated. For additional information, refer to the first-round examination report.

4.4 F-04 Insufficient security documentation

Initial finding description (round 1 examination – March 2022)

Key	F-04
Title	Insufficient security documentation
Requirement ID(s)	24.1.20
Requirement(s)	Software development security documentation includes: » **a description of the physical, procedural, personnel, and other security measures necessary to protect and ensure the integrity of the design and implementation of the software in its development environment (a);** » **evidence that the security measures provide the necessary level of protection to preserve the integrity of the software (b).**
Severity	Moderate
Rationale	The examiners noted the following: » A large amount of documentation had been made available, including documented procedures, architecture and design documents, features documentations and protocol specifications. » However, the examiners noted that, although the e-voting cryptographic protocol had been extensively documented, reviewed and tested, many other security-relevant aspects of the e-voting system had not yet been documented. In particular, the examiners had noted the absence of threat models, baseline architecture and design principles, coding guidelines, security testing procedures, code attestation procedures, and moreover, the absence of a central document that would tell third-party how the e-voting system, in general, is secured[8].
Recommendations	R-08 - Establish a central document that describes how security assurance in the e-voting system is obtained (e.g., e-voting security whitepaper).

Table 11 - Finding F-04 (Insufficient security documentation)

[8] Here, the examiners make (and made) a distinction between the security of the e-voting system (the sum of all things put together, in the eye of the user) and the security of the e-voting protocol itself, which are considered two distinct, although highly interconnected, concepts.

Follow-up examination (round 2 – October 2022)

Evidence

52. The following evidence was provided:

- Document: Security whitepaper – Software development – e-voting [20]
- On-site interview

Assessment

53. On the recommendation (R-08):

- The examinee produced a security whitepaper for the e-voting system [20] in response to the first round examination recommendations. The document reproduces the topic structure set forth in the OWASP's software assurance maturity model (SAMM) [23] and explains how the e-voting system's development team achieves the specified objectives for each function (i.e., governance, design, implementation, verification, operations).
- After reviewing the document, the examiners' first conclusion was that the recommendation had not been met satisfyingly. In particular, the document did not include the recommended content [9, Para. 51] but rather focused on describing the e-voting development process.
- The examiners shared their concern and conclusions with the examinee during a workshop. A constructive point was voiced: the legitimacy of the development team to produce the requested document was questioned. It appears that the content, as it had been specified in the first examination report, would require extensive contributions from many parties involved in the project and not only development (i.e., business stakeholders, security management, software development, Swiss Post infrastructure operations, both for employees and for live operations,). The document could therefore not be solely produced as a development team initiative.
- The examiners agreed, and an adaptation of the initial recommendation was discussed: the development team will set up a new page in the documentation website [25], which will centralize pointers to security-relevant documentation about the e-voting system. The team will initiate the work by inventorying and listing its own documents, thus opening the way for other e-voting divisions to also contribute with their security-related documentation.
- While the examiners recognize that this approach will not resolve the initial concern for information spread (security-relevant information being spread across dozens of documents), it will nonetheless establish a starting point for centralizing all pointers to security-relevant information on the e-voting project collaboratively.

54. Based on the evidence and the proposed action plan shown to the examiners, the implementation of recommendation R-08 is considered to be in progress and aimed towards completion.

4.5 F-05 Insufficient quality control over security attestation operations

Initial finding description (round 1 examination – March 2022)

Key	F-05
Title	Insufficient quality control over security attestation operations
Requirement ID(s)	24.5[9]
Requirement(s)	**Regular and objective checks are carried out to ensure that the processes carried out and the associated work products comply with the description of the processes, standards and procedures to be implemented (a).** » Deviations are followed up until they are corrected (b).
Severity	Moderate
Rationale	The examiners noted the following: » The examinee has put in place a robust quality assurance system, which ensures most, if not all, its operations and procedures are documented, and regularly improved. » It has implemented a large set of controls, including tools and processes, to produce security attestations and overall trust over the e-voting system. These include, among others, source code review tools, software composition analysis tools, public audits, external penetration testing, bug bounties, etc. » While these tools and controls are aimed at producing assurance about the security of the e-voting system, the effectiveness and correctness of operation of these tools was not governed by the quality assurance process in place.
Recommendations	R-09 - Establish procedures to confirm, review and improve the correct operation of security attestation measures, in particular automated measures.

Table 12 - Finding F-05 (Insufficient quality control over security attestation operations)

Follow-up examination (round 2 – October 2022)

Evidence

55. The following evidence was provided:

- Document: Security whitepaper – Software development – e-voting [20]
- On-site interview

[9] A typo in the first-round examination report incorrectly mentioned requirement 24.1.15 (configuration management system documentation) in addition to requirement 24.5 (quality assurance).

Assessment

56. On the recommendation (R-09):

- The examiners note that the examinee has not yet implemented the recommended controls [9, Para. 54] in its quality assurance processes.
- The examinee anticipates that completing the implementation of recommendations R-10 (design security reviews) and R-11 (runtime testing) may be a pre-requisite to address this recommendation, as it is in the process of acquiring and integrating said tools and processes, respectively.

57. Based on the evidence shown to the examiners, the implementation of recommendation R-09 is considered to be incomplete.

Additional remark

58. The examiners noted that recommendation R-09 could be interpreted as general lack of quality control in the e-voting project and/or lack of quality control of the deliverables. This would be an incorrect interpretation of this recommendation, considering that the examiners qualified the quality control system put in place by Swiss Post as exemplary [9, Para. 94], [9, Para. 95].

The quality assurance requirement, as mandated by the Federal Chancellery, instructs the examinee to carry out regular and objective assessments to ensure that the processes carried out as part of the e-voting development comply with their original intent [9, App. 24.5].

In the spirit of this mandate, recommendation R-09 is not about verifying the deliverable itself but about verifying that the security tools used to validate the deliverable, in particular, do work as intended and as expected.

4.6 F-06 Insufficient security testing and attestation

Initial finding description (round 1 examination – March 2022)

Key	F-06
Title	Insufficient security testing and attestation
Requirement ID(s)	17.2
Requirement(s)	An analysis must be made of the test coverage. This includes evidence that: » the tests defined in the test documentation match the functional specifications of the interfaces (a); » **all interfaces have been fully tested (b).**
Severity	Moderate
Rationale	The examiners noted the following:

	» The examinee demonstrated strong evidence of security assurance along the entire lifecycle of the e-voting protocol. » However, the same coverage could not be observed for other areas of the e-voting system, especially parts designated as untrusted components per the e-voting protocol specification. » In particular, the examiners noted limited security assessment activities prior to entering the development (coding) phase and a lack of runtime security testing (prior to deployment).
Recommendations	R.10 - Establish procedures to review and/or validate design proposals generated in response to change requests, prior to entering the coding phase. R.11 - Establish or improve runtime/dynamic application security testing procedures executed prior to release. For relevant components, extend these procedures with additional fuzz testing.

Table 13 - Finding F-06 (Insufficient security testing and attestation)

Follow-up examination (round 2 – October 2022)

Evidence

59. The following evidence was provided:

- Document: Security whitepaper – Software development – e-voting [20]
- Document: Secure design and coding guidelines [22]
- On-site interview

Assessment

60. On the first recommendation (R-10):

- Although architects are instructed to ensure that all submissions flagged with architectural risk adhere to the rules set forth in the security guidelines [22], the examinee has not yet formalized the criteria on which work items are submitted to such verification. It is currently left to the judgment of the developer (informal process).
- An initiative aims at inducing formal security controls into the design approval process. This will improve the existing criteria to qualify a work item as ready for development (definition of ready).

61. Based on the evidence and the proposed action plan shown to the examiners, the implementation of recommendation R-10 is considered to be in progress and aimed towards completion.

62. On the second recommendation (R-11):

- The examinee is currently evaluating and testing several commercial runtime application security testing (DAST) solutions. Their deployment is expected during in the first half of 2023.
- It is also deploying an interactive application security testing (IAST) solution at the organization level and evaluating it for use as part of its e-voting security testing operations.

63. Based on the evidence and the proposed action plan shown to the examiners, the implementation of recommendation R-11 is considered to be in progress and aimed towards completion.

4.7 Summary of recommendations and statuses

Key	OEV key(s)	Finding	Recommendation	Status
R-01	24.1.1	F-01	Formalise the integration of security development lifecycle guidance or best practices into the e-voting system's development process.	Completed
R-02	24.1.1	F-01	Ensure e-voting system personnel and stakeholders have received adequate role-based training on secure systems engineering.	Completed
R-03	24.1.1	F-01	Integrate threat modelling, or equivalent, in early stages of the development process.	In progress
R-04	24.1.1	F-01	Establish a baseline set of security principles or rules for each phase of the development lifecycle (e.g., requirements, architecture and/or design, coding, testing, build, deployment, etc.).	Completed
R-05	24.1.1	F-02	Establish a security champion program and appoint a champion in each e-voting system development team.	Completed
R-06	24.1.1	F-03	Establish a reference threat model for the use of third-party components in the e-voting system, maintained with the status of implementation of chosen controls and countermeasures.	Pending
R-07	24.1.1	F-03	Establish a security vetting process for the selection of new third-party components, and the review of existing ones, embedded in the e-voting system.	In progress
R-08	24.1.20	F-04	Establish a central document that describes how security assurance in the e-voting system is obtained (e.g., e-voting security whitepaper).	In progress

R-09	24.5	F-05	Establish procedures to confirm, review and improve the correct operation of security attestation measures, in particular automated measures.	Pending
R.10	17.2	F-06	Establish procedures to review and/or validate design proposals generated in response to change requests, prior to entering the coding phase.	In progress
R.11	17.2	F-06	Establish or improve runtime/dynamic application security testing procedures executed prior to release. For relevant components, extend these procedures with additional fuzz testing.	In progress

Table 14 - Summary of recommendations and statuses

4.8 Summary of findings and statuses

Key	OEV key(s)	Finding	Severity	Recommendation	Remediation status
F-01	24.1.1	Insufficient integration of security in the software development lifecycle	Moderate	R-01, R-02, R-03, R-04	In progress
F-02	24.1.1	Conflicting / ambiguous attribution of security responsibilities	Moderate	R-05	Resolved (closed)
F-03	24.1.1	Insufficient protection from risky third-party components	Moderate	R-06, R-07	In progress
F-04	24.1.20	Insufficient security documentation	Moderate	R-08	In progress
F-05	24.5	Insufficient quality control over security attestation operations	Moderate	R-09	In progress
F-06	17.2	Insufficient security testing	Moderate	R-10, R-11	In progress

Table 15 - Summary of findings and statuses

5 Examination results

5.1 Scope 2a (software development process)

Development process and lifecycle requirements

Key	Requirement	Finding(s)	Decision
24.1.1	A life cycle model is defined. The life cycle model: » is used for the development and maintenance of the software (a); » provides for the necessary controls during the development and maintenance of the software (b); » is documented (c).	F-01 F-03	P.MISS
24.1.2	A list must be made of the development tools used and configuration options chosen for the use of each development tool.	-	PASS
24.1.3	The documentation for the development tools includes: » a definition of the development tool (a); » a description of all conventions and directives used in the implementation of the development tool (b); » a clear description of the significance of all configuration options for using the development tool (c).	-	PASS
24.1.4	The implementation standards to be applied must be specified.	-	PASS

Table 16 - E-voting requirements: lifecycle

Software security documentation requirements

Key	Requirement	Finding(s)	Decision
24.1.20	Software development security documentation includes: » a description of the physical, procedural, personnel, and other security measures necessary to protect and ensure the integrity of the design and implementation of the software in its development environment (a); » evidence that the security measures provide the necessary level of protection to preserve the integrity of the software (b).	F-04	P. MISS

Table 17 - E-voting requirements: software security documentation

Quality assurance requirements

Key	Requirement	Finding(s)	Decision
24.5	Regular and objective checks are carried out to ensure that the processes carried out and the associated work products comply with the description of the processes, standards and procedures to be implemented (a). Deviations are followed up until they are corrected (b).	F-05	MISS

Table 18 - E-voting requirements: quality assurance

Configuration management system requirements

Key	Requirement	Finding(s)	Decision
24.1.14	The software is provided with a unique identification.	-	PASS
24.1.15	The configuration management documentation includes: » a description of how configuration items are identified (a); » a configuration management plan describing how the configuration management system will be used in the development of the software and the procedures that will be followed for the adoption of changes or new elements (b); » evidence that the procedures for adoption provide for adequate review of changes for all configuration items (c).	-	PASS
24.1.16	The configuration management system: » uniquely identifies all configuration items (a); » provides automated measures to ensure that only authorised changes are made to configuration items (b); » supports the development of the software through automated procedures (c); » ensures that the person responsible for accepting the configuration item is not the same person who developed it (d); » identifies the configuration items that make up the security functions (e); » supports verification of all changes to the software using automated procedures, including logging of the author and the date and time of the change (f); » provides an automated method for identifying any configuration items that are affected by a change to a particular configuration item (g);	-	PASS

	» can identify the version of the source code on the basis of which the software is generated (h).		
24.1.17	All configuration items are inventoried in the configuration management system.	-	PASS
24.1.18	The configuration management system is used in accordance with the configuration management plan.	-	PASS
24.1.19	A configuration list is created that contains the following items: » the software, » evidence of the checks required to ensure security compliance, » the parts that make up the software, » the source code, » the commit history[10], » reports on security flaws and on the status of their correction (a). For each element relevant to security functions, the developer is named (b). Each element is uniquely identified (c).	-	PASS

Table 19 - E-voting requirements: configuration management system

Testing requirements

Key	Requirement	Finding(s)	Decision
17.1	The functions relevant to the security of the system (security functions) are tested. The tests are documented with test plans and expected and actual test results. (a) The test plan (b): » specifies the tests to be performed; » describes the scenarios for each test, including any dependencies on the results of other tests. The expected results must show the results that are expected if the test is successfully executed. (c) The actual results must be consistent with expected results. (d)	-	PASS

[10] The requirement for a commit history was added in the revised ordinance.

17.2	An analysis must be made of the test coverage. This includes evidence that: » the tests defined in the test documentation match the functional specifications of the interfaces (a); » all interfaces have been fully tested (b).	F-06	P. MISS
17.3	An analysis must be made of the depth of testing. This includes evidence that: » the tests defined in the test documentation match the subsystems related to security functions and modules that play a role in ensuring security (a); » all subsystems related to the security functions mentioned in the specifications have been tested (b); » all modules that play a role in ensuring security have been tested (c).	-	PASS
25.13.3	The integration tests cover all modules.	-	PASS
25.13.4	The software tests cover all modules.	-	PASS

Table 20 - E-voting requirements: testing

Transparency requirements

Key	Requirement	Finding(s)	Decision
8.13[11]	Known flaws and the need for action associated with them are communicated transparently. **: identified as 8.12 in the round 1 examination criteria.*	-	PASS

Table 21 - E-voting requirements: transparency

Systematic correction of flaws requirements

Key	Requirement	Finding(s)	Decision
24.4.1	Processes are defined for the correction of flaws. The processes include: » documentation of specific aspects, in particular with regard to the traceability of flaws for all versions of the software, and of the methods used to ensure that system users have information on flaws, corrections and possible corrective actions (a); » the obligation to describe the nature and impact of all security flaws, information on the status of work to find a solution and the corrective measures adopted (b);	-	PASS

[11] Identified as requirement key 8.12 in the round 1 examination report, the requirement remains unchanged.

	» a description of how system users can make reports and enquiries about suspected flaws in the software known to the software developers (c); » a procedure requiring a timely response and automatic dispatch of security flaw reports and appropriate corrective actions to registered system users who may be affected by the flaw (d).		
24.4.2	A process is defined for handling reported flaws (a). This process ensures that all reported and confirmed flaws are corrected and that the procedures for correction are communicated to system users (b). It provides for arrangements to ensure that the correction of security flaws does not give rise to new security flaws (c).	-	PASS
24.4.3	Policies must be defined for the reporting and correction of flaws. These include: » instructions on how system users can report suspected security flaws to the developer (a); » instructions on how system users can register with the developer to receive reports of security flaws and the corrections (b); » details of specific contact points for all reports and inquiries on security issues concerning the software (c).	-	PASS

Table 22 - E-voting requirements: systematic correction of flaws

5.2 Summary of pending or incomplete requirements

Key	Requirement	Decision
24.1.1	A life cycle model is defined. The life cycle model: » is used for the development and maintenance of the software (a); » provides for the necessary controls during the development and maintenance of the software (b); » is documented (c).	P.MISS
24.1.20	Software development security documentation includes: » a description of the physical, procedural, personnel, and other security measures necessary to protect and ensure the integrity of the design and implementation of the software in its development environment (a); evidence that security measures provide the necessary level of protection to preserve the integrity of the software (b).	P.MISS
24.5	Regular and objective checks are carried out to ensure that the processes carried out and the associated work products comply with the description of the processes, standards and procedures to be implemented (a). Deviations are followed up until they are corrected (b).	MISS

17.2	An analysis must be made of the test coverage. This includes evidence that: » the tests defined in the test documentation match the functional specifications of the interfaces (a); all interfaces have been fully tested (b).	P.MISS

Table 23 – Summary of pending or incomplete requirements (summary)

5.3 Summary of remaining actions / recommendations

Key	OEV key(s)	Recommendation	Status
R-03	24.1.1	Integrate threat modelling, or equivalent, in early stages of the development process.	In progress
R-06	24.1.1	Establish a reference threat model for the use of third-party components in the e-voting system, maintained with the status of implementation of chosen controls and countermeasures.	Pending
R-07	24.1.1	Establish a security vetting process for the selection of new third-party components, and the review of existing ones, embedded in the e-voting system.	In progress
R-08	24.1.20	Establish a central document that describes how security assurance in the e-voting system is obtained (e.g., e-voting security whitepaper).	In progress
R-09	24.5	Establish procedures to confirm, review and improve the correct operation of security attestation measures, in particular automated measures.	Pending
R.10	17.2	Establish procedures to review and/or validate design proposals generated in response to change requests, prior to entering the coding phase.	In progress
R.11	17.2	Establish or improve runtime/dynamic application security testing procedures executed prior to release. For relevant components, extend these procedures with additional fuzz testing.	In progress

Table 24 - Summary of recommendations and statuses

6 References

[1] "Reorienting eVoting and ensuring stable trial operation," *www.egovernment.ch*. https://www.egovernment.ch/en/umsetzung/schwerpunktplan/vote-electronique/ (accessed Oct. 21, 2021).

[2] Swiss Federal Chancellery, Political Rights Section, "Redesign and relaunch of trials - Final report of the Steering Committee Vote électronique (SC VE)." Nov. 30, 2020. Accessed: Dec. 06, 2021. [Online]. Available: https://www.bk.admin.ch/dam/bk/en/dokumente/pore/Final%20report%20SC%20VE_November%202020.pdf.download.pdf/Final%20report%20SC%20VE_November%202020.pdf

[3] Swiss Federal Chancellery, Political Rights Section, "Partial revision of the Ordinance on Political Rights and total revision of the Federal Chancellery Ordinance on Electronic Voting (Redesign of Trials)." Apr. 28, 2021. Accessed: Dec. 06, 2021. [Online]. Available: https://www.bk.admin.ch/dam/bk/en/dokumente/pore/Explanatory%20report%20for%20consultation%202021.pdf.download.pdf/Explanatory%20report%20for%20consultation%202021.pdf

[4] Swiss Federal Chancellery, "Federal legislation." https://www.bk.admin.ch/bk/en/home/politische-rechte/e-voting/versuchsbedingungen.html (accessed Oct. 21, 2021).

[5] Swiss Federal Chancellery (FCh) - Political Rights section, "Audit concept for examining Swiss Internet voting systems - v1.3." May 18, 2021.

[6] Swiss Federal Chancellery, "Federal Chancellery ordinance on electronic voting (OEV)." Apr. 28, 2021. [Online]. Available: https://www.bk.admin.ch/dam/bk/en/dokumente/pore/OEV_draft%20for%20consultation%202021.pdf.download.pdf/OEV_draft%20for%20consultation%202021.pdf

[7] Swiss Federal Chancellery, Political Rights Section, "Partial revision of the Ordinance on Political Rights and total revision of the Federal Chancellery Ordinance on Electronic Voting (Redesign of Trials)." May 25, 2022. Accessed: Dec. 06, 2021. [Online]. Available: https://www.newsd.admin.ch/newsd/message/attachments/71705.pdf

[8] Swiss Federal Chancellery, "Federal Chancellery ordinance on electronic voting (OEV)." May 25, 2022. [Online]. Available: https://www.fedlex.admin.ch/eli/cc/2022/336/en

[9] A. Fontes, "Examination of the Swiss Internet voting system - Audit scope 2a - Development processes." Mar. 22, 2022. [Online]. Available: https://www.admin.ch/gov/en/start/documentation/media-releases.msg-id-88085.html

[10] Swiss Post, "Post Audit Response to examination report by SCRT – Scope 2a Software (Internal document)." Jul. 29, 2022.

[11] Swiss Federal Chancellery (FCh) - Political Rights section, “Audit concept for examining Swiss Internet voting systems - v1.4.” Apr. 12, 2022. [Online]. Available: File reference: 431.0-2/5/12/16

[12] Payment Card Industry, “Secure software lifecycle (Secure SLC) requirements and assessment procedures - PCI-SSF v1.0.” Jan. 2019.

[13] Center for Internet security, “CIS Controls Version 8,” *CIS*. https://www.cisecurity.org/controls/v8/

[14] Software assurance forum for excellence in code (SAFECode), “Fundamental practices for secure software development - 3rd ed.” Mar. 2018.

[15] Watson, Colin, Lynch, Aidan, Coblentz, Nick, Keary, Eoin, and Deleersnyder, Seba, “SAMM Assessment toolbox v1.5 final.” OWASP, 2009. [Online]. Available: https://github.com/OWASP/samm/tree/master/Supporting%20Resources/v1.5

[16] Software assurance forum for excellence in code (SAFECode), “Managing security risks inherent in the use of third-party components.” 2017.

[17] Swiss Post, “Application security radard@e-voting - OWASP-SAMM assessment Q1-2022 (internal document).” Apr. 19, 2022.

[18] Swiss Post, “SAMM Assessment EV22-Ergebnisse Q1-2022 (Internal document).” Jul. 29, 2022.

[19] Swiss Post, “Checkliste fur neuer Mitarbeiter (internal document).” May 10, 2022.

[20] Swiss Post, “Security Whitepaper - Software Development E-Voting (Internal document).” Jul. 29, 2022.

[21] Swiss Post, “Training records - e-voting (Internal document).” Oct. 05, 2022.

[22] Swiss Post, “Hitchhiker’s Guide to Application Security (Internal document).” 2022.

[23] OWASP, “Software assurance maturity model (SAMM),” *OWASP SAMM*, Sep. 14, 2022. https://owaspsamm.org/ (accessed Dec. 08, 2021).

[24] L. Kohnfelder and P. Garg, “The threats to our products,” *Microsoft Interface Microsoft Corp.*, vol. 33, 1999.

[25] Swiss Post, “Swiss Post e-voting documentation (home),” *GitLab*. https://gitlab.com/swisspost-evoting/e-voting/e-voting-documentation (accessed Nov. 21, 2021).