Log Analysis of Estonian Internet Voting 2013–2015*

Sven Heiberg[1], Arnis Parsovs[2,3], and Jan Willemson[4]

[1]Smartmatic-Cybernetica Centre of Excellence for Internet Voting, Estonia
[2]Software Technology and Applications Competence Centre, Estonia
[3]University of Tartu, Institute of Computer Science, Estonia
[4]Cybernetica, Estonia

December 10, 2015

Abstract

In this report we describe our efforts in analysing log files produced by the Estonian i-voting system in the KOV2013, EP2014 and RK2015 elections in combination with other information available, so as to detect attacks against the i-voting system, detect system malfunctions and study voter behaviour.

1 Introduction

The RK2011 elections were the turning-point in Estonian i-voting. The share of votes cast over the Internet reached the critically high 24.3% mark [2] and the case of proof-of-concept election rigging malware demonstrated by a student [3, Section 3.1] brought to light a discussion about attacks that can be executed against the Estonian i-voting system and the ability of the Estonian National Electoral Committee (NEC) to detect such attacks. As a result, the i-voting protocol was extended by adding a vote verification scheme [4] that provides cast-as-intended verification for Estonian i-voting, and an initiative was established to perform an in-depth analysis of the logs produced by i-voting servers and other information available to the NEC, in order to detect system malfunctions and attacks against the i-voting system, and study voter behaviour.

In this report we describe a log monitoring and analysis solution, which was applied in the KOV2013, EP2014 and RK2015 elections, and report on the results obtained from these elections.

2 Log monitoring

2.1 Estonian Internet voting scheme

The basic Internet voting scheme used in Estonia follows the double-envelope postal voting system where the inner envelope is replaced by encryption and the outer envelope by a digital signature (see [3] for a more detailed description). For cryptographic operations, each voter can use either smart card-based eID tools (ID card, Digi-ID) or cellphone SIM card-based Mobile-ID. The voter is supplied with the official i-voting client application (IVCA) and she can use it to download the candidate list and cast her vote to the server. Since the 2013 elections it has also been possible to verify one's vote using a mobile device [4]. In case the Internet voter feels coerced, she can resubmit her vote via the Internet or in a polling station during the advance voting period.

*A shorter version of this paper is published in the Proceedings of the 5th International Conference on E-voting and Identity, Springer LNCS 9269 [1].

The three protocols implemented by the i-voting system – voting with a smart card, voting with Mobile-ID and verification – are defined by finite-state machines. The transitions between the states generate log messages. For example, Figure 1 displays the protocol for retrieving the candidate list with a smart card-based eID tool (other UML diagrams have been included in Appendix A). After TLS authentication to the Vote Forwarding Server (VFS) has succeeded, a unique session identifier *sid* is generated. The *sid* is used throughout the voting session to identify log messages belonging to this protocol run. Before proceeding to eligibility checks and candidate list retrieval, the IP address, HTTP User-Agent, personal code and client certificate of the voter are logged. The protocol proceeds by determining the eligibility of the voter, checking the revoting status at the Vote Storage Server (VSS) and returning the candidate list to IVCA. Each of those steps is logged accordingly. Candidate list retrieval is later followed by vote casting where the same *sid* is submitted by the IVCA.

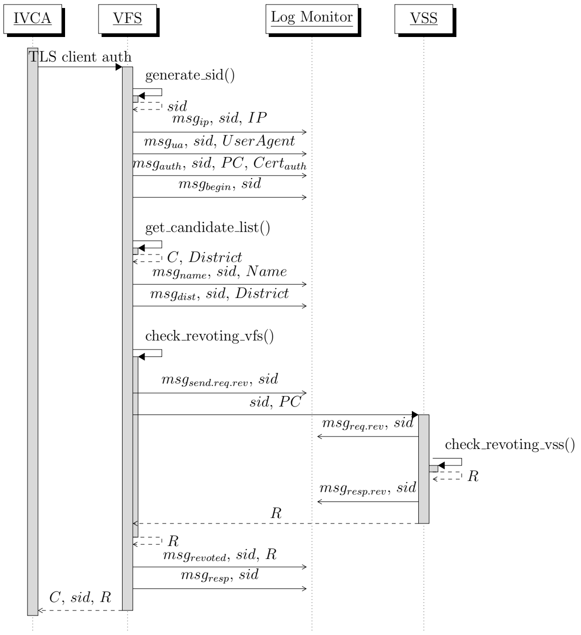


Figure 1: Logs generated on candidate list retrieval (ID card)

During the i-voting period, a large amount of log entries is produced (e.g. in 2013, 4,086,512 messages). Since it is not feasible for election officials to manually review every log entry, a solution was required to process the produced audit trail and generate a meaningful summary report that could be used to assess the current state of the i-voting system and perform informed decisions based on it. For example, an unusually high system load could signal a possible bug in the server software or an ongoing denial-of-service attack. A sudden increase in the number of unfinished voting sessions could be caused by a bug in the i-voting software or an attack being carried out on Internet voters, etc.

A log monitor has been introduced to the architecture. The monitor is connected to the VFS and VSS receiving copies of log messages in quasi real-time using the `rsyslog` utility with UDP as the transport protocol.

The analysis software consists of three main components: a log processing engine that parses every log entry and updates the database; a database engine where log information is stored in a relational model; and a web front-end that performs analysis on the data from the database and shows the result to the election officials. In the following subsections we describe every component in detail.

2.2 Log Processor

The log processing engine is a Python program that parses every log line and by using regular expressions tries to match the line against the list of defined patterns. Useful information from the log entry is extracted and inserted into the database. Every log entry that cannot be strictly matched against the list of expected entries is written into the database as an incident requiring manual inspection by an election official. Usually such incidents are exceptions raised by the i-voting software. The log processor also performs a basic log entry order check (see Section 2.7.2) and raises an incident if it detects an incorrect order.

2.3 Database

For data storage we use the open-source relational database management system MySQL. Since transaction support is not required, the data is stored and indexed using the MyISAM storage engine.

The database table structure is shown in Figure 2. The central table is the `sessions` table, which contains data describing a voting session. The `verifications` table contains information about vote verification requests. Verification requests can be linked to voting sessions through the `vote_id` field.

The `incidents` table stores incidents that have been logged by the log processor. The incidents are linked to the `incident_response` table, which stores incident resolutions created by election officials.

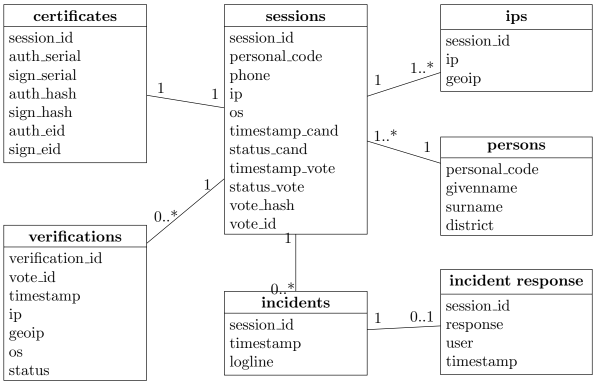


Figure 2: Database table structure

2.4 Web Front-end

In a way, the web front-end is the core of the monitoring solution, since its task is to analyse the data stored in the database and produce an output that is meaningful for election officials. The web front-end is connected to the Internet and is accessible to election officials after they have authenticated themselves using an Estonian ID card.

The functionality of the web front-end can be divided into three parts. Next we will describe each part in detail.

2.4.1 Healthiness of the System

This part implements basic i-voting system healthiness oversight. Several parameters such as system load, memory usage and free disk space are collected every minute from VFS and VSS. These parameters are displayed in the web interface and if the most recent data record is older than one minute the web interface raises an alert. A watchdog that checks the responsiveness of the OCSP server maintained by the Estonian National Certificate Authority (NCA) is implemented similarly. Finally, the status of the log processor's system process is also shown.

2.4.2 Incident Management

The incident management view provides a list of incidents that have been logged by the log processor. Every incident is linked to a particular voting session and to a log entry that caused it. Once the election official reviews the incident, she can submit a resolution text. After reviewing the incident the status of the incident is updated to "handled".

The web front-end implements a functionality that supports the investigation of the circumstances of an incident by providing context around the incident. Throughout the web interface any voting session can be connected to all other i-voting sessions that involve any associated parameter such as personal code, IP address or phone number. Using this capability, the election official can check, for example, whether the IP address involved in the incident is also involved in other incidents and whether the person whose voting session is linked to the incident was eventually able to cast a vote.

The IP address view is enhanced with geographical information using the MaxMind GeoLite City database [5]. All entries containing the IP address are extracted from Apache log files and shown in the IP view. The voting session view displays data from the database and extracts all log file entries that are connected to the voting session.

2.4.3 Data Analysis

This part implements a list of predefined SQL queries. First it generates basic descriptive statistics about the votes cast and verified. Then it executes a list of SQL queries that search the data for anomalies that could indicate that an attack is being carried out against the i-voting system.

2.5 Specification-based log analysis

The relative simplicity of voting and verification protocols makes it feasible to apply the specification-based approach to monitoring where manually developed specifications are used to characterise legitimate program behaviours. Sessions that describe valid protocol runs and end with a successful result or an acknowledged error state are generally not interesting for detailed analysis. These sessions are white-listed, they may become the subject of analysis if an external condition characterises them as part of some bigger pattern – e.g. somebody re-voting a number of times over a certain threshold.

Associated with each session is a set of data which should be consistent within the session and/or across sessions. If certain conditions are not met the session is labelled for further analysis.

2.6 Normality Profile

First we briefly define the normality profile of i-voting and then we consider anomaly as an inverse of this profile. In the next section we discuss possible attacks and their associated patterns that could be found in the data.

1. The voting and verification session creates only expected log entries.
2. The state of voting and verification session changes in the expected order.
3. The voting session ends with a successfully cast vote.
4. The verification session ends with a successfully verified vote.
5. The voter does not experience any difficulties related to voting or verifying.
6. The voting session is completed in a few minutes.
7. The encrypted vote is signed with the same eID tool that was used for authentication.
8. The IP address and the OS of the voter do not change throughout the voting session.
9. Not too many voters share the same voting IP address.
10. The vote encryption is unique.
11. The overall percentage of revoters is small.
12. The voter revotes only a few times.
13. If the voter revotes she is using the same eID tool (ID card, Digi-ID or Mobile-ID).
14. If the voter revotes using the same eID tool, the certificates are the same.
15. If the voter revotes she is revoting from the same or geographically close IP address.
16. A new voting session starts after the voter's previous voting session is finished.
17. The vote is verified from a single IP address.
18. The voter's votes are verified from a few IP addresses.
19. Not too many voters share the same IP to verify the vote.
20. When an i-voter casts her vote for the first time, she is not receiving a response that states that she is revoting.
21. When the non-i-voter goes to a polling station on the election Sunday, she is not prohibited from voting with the claim that she has already i-voted.
22. The i-voting results do not deviate too much from the paper voting results.
23. General statistics do not deviate too much between elections.
24. Other irregularities are handled and explained.

2.7 Anomaly Patterns

Based on the normality profile described above, we define anomaly patterns that should be matched against the data and discuss considerations that should be applied when analysing such anomalies.

Since most of the anomaly patterns described also match actions that can be explained by legitimate voter behaviour, some of these events may not be considered anomalous. However, if the total number of events for a particular pattern significantly changes between elections, that should be considered an anomaly and investigated.

In the following sections of this report we will analyse data from the KOV2013, EP2014 and RK2015 elections using the anomaly patterns defined below.

2.7.1 Unexpected log entries

A voting session that creates an unexpected log entry is a strong indicator that some of the i-voting components are misbehaving. Such an event is marked as an incident and has to be manually investigated by an election official (see Section 2.4.2).

2.7.2 Incorrect session state change

When the log processor parses log entries, before updating session status, the log processor checks whether the new status is the expected next status of the current session status registered in the database. If this is not the case the log processor registers an incident. Such an incident would indicate log file inconsistency or some other system malfunction.

2.7.3 Unsuccessful voting sessions

Sessions that fail with a known error code (such as "ineligible voter") are not of great interest. Similarly, voting sessions that fail with an unknown error code will be logged as incidents and will be reviewed manually by an election official. However, sessions whose status corresponds to any intermediate processing step are of great interest. For example, if the voting session has the status "the i-vote has been received", but the status is not eventually updated to "successfully issued vote identifier" and no incident is registered, then this would indicate either a logging system failure or a server process crash caused by a software bug or by an exploitation attempt.

A special category of unsuccessful voting sessions is unfinished voting sessions, i.e. cases where a vote submission request does not follow. The cause can be a software bug in IVCA that prevents the voter from successfully casting an i-vote, a disenfranchisement attack executed by malware on the voter's computer, or an attacker who has obtained access to the voter's eID tool and is checking whether the voter has already i-voted in order to revote and escape the detection mechanism described in Section 2.7.20.

On the other hand the voter may be just verifying whether she is eligible to vote and which candidates are running in the district she is registered to. An additional challenge is that the i-voting protocol does not have a timeout enforcement and therefore we cannot be certain whether the IVCA has been closed or is the session still in progress.

2.7.4 Unsuccessful verifications and verification sessions

The purpose of the vote verification scheme described in [4] is to detect a large-scale vote manipulation attack if such is executed against voters' voting devices. Even if only a small part of voters verify their vote the probability of a large-scale attack going undetected is minimal [4, Section VI] (considering also the attacks described in Section 2.7.11 and 2.7.18).

The voters who do not succeed in verifying their vote are expected to contact the NEC for investigation. The are mainly two cases of verification failures that can occur.

The first case of failure is when the vote cannot be decrypted or it is decrypted to a candidate that the voter did not choose. Such a failure would be a strong indicator of a critical software bug or vote-changing malware on the voter's voting device. The occurrence of such a failure can be detected by the NEC only if the voter contacts the NEC after experiencing such a failure.

The second case is when the vote requested cannot be found on the election server. Such a failure is logged on the server side and the error message is shown to the voter with contact information of the NEC. The error can be caused by several reasons. This can occur if the verification time (usually 30 minutes) or the number of allowed verifications (3 verifications) has been exceeded by the voter, the voter has revoted, thus overwriting the vote being verified, the voter is trying to verify the QR code of a vote that has not been issued for the ongoing elections or the vote identifier is being brute-forced by an attacker.

2.7.5 Support requests handled by the NEC support centre

A large part of irregularities in the voting process may not be visible on the server side and can become known only if the end-user contacts the NEC support centre. A large part of support requests are expected to be general inquiries about the voting process, but some of them might also inform the NEC about serious bugs or even security issues.

2.7.6 Voting sessions too slow

A voting session that is too slow could indicate that the IVCA is being reverse engineered or an attack is being developed as it was in the RK2011 proof-of-concept malware case [3, Section 3.2]. However, a voting session can be slow also because of the voter's completely legitimate behaviour, for example, if the voter is in the process of i-voting and is interrupted by other tasks, and finishes the voting several hours after initiating the voting process.

2.7.7 Vote signed with a different eID tool

The i-voting protocol allows, for example, for the voter to authenticate using an ID card and submit a vote signed by the Mobile-ID eID tool. However, that would be an anomaly since the official IVCA does not implement such a feature and there is no reason for the voter to use two eID tools during one voting session.

2.7.8 IP address or OS change in the middle of a voting session

An i-voting session that uses an ID card consists of two HTTP requests. In the first request the IVCA obtains the candidate list and in the second request the IVCA submits the encrypted and signed vote. If Mobile-ID is used for i-voting, the protocol consists of additional HTTP requests that perform authentication and signing according to the Mobile-ID protocol [6, Section 2].

Voter's IP address or OS[1] change in the middle of a voting session might indicate voting session hijacking. Although we do not see the benefit or the flaw that would allow hijacking the i-voting session, we believe that the detection of such an anomaly is advisable.

Note that an IP address change could happen also for a completely legitimate reason, such as the voter switching Internet connections in the process of i-voting.

2.7.9 IP address shared by several voters

If several voters use the same IP address to cast a vote in a short time frame, that could indicate that collective voting is being performed using a single voting device or that the votes are cast by a single person using the eID tools of other persons.

Several voters can also legitimately use a single IP if they are, for example, voting from a large organisation where a shared connection is used to access the Internet.

2.7.10 Non-unique vote encryption

The IVCA produces an encrypted vote by encrypting the selected candidate number along with the election identifier using the RSA-OAEP encryption scheme, which adds random padding to achieve semantic security. Therefore, the i-voting servers should not receive any duplicate votes.

Several encrypted votes sharing the same hash value could indicate either a randomness failure in the IVCA (as it was in the 2013 parliamentary elections in Norway [7]), vote manipulation malware that uses hard-coded version of encrypted vote, or a ballot copying attack [8].

[1] The voter's OS is obtained from the IVCA "User-Agent" header set in the HTTP request.

2.7.11 Large percentage of revoters

In order to prevent vote selling and coercion, voters can change their i-vote by casting another i-vote. Throughout the previous elections (KOV2005, RK2007, EP2009, KOV2009 and RK2011) the revoter proportion was 3.9%, 2.6%, 1.55%, 2.27% and 3.11%, respectively [2][2].

A sudden increase in the proportion of revoters should be considered an anomaly. This could indicate large-scale coercion that forces voters to revote or malware installed on the voting devices that revotes using the voter's eID tool connected to the device, thus escaping detection by the vote verification scheme [4, Section V.E].

An increase in the revoter ratio could also have a legitimate reason. For example, a significant political scandal during the seven-day advance i-voting period could convince voters to change their minds and revote for a different candidate.

2.7.12 Voters revoting many times

Voters revoting many times are anomaly. It is unlikely for a voter to change her political preference more than a few times. Instead it could be a person trying to reverse-engineer the IVCA or an attacker intensively testing the attack. Finally, revoting could be used also as a peculiar form of denial-of-service attack. Some persons involved in the organisation of i-voting can cause a large number of revotes because they are testing and demonstrating the i-voting system.

In RK2011 there was a case of a voter casting more than 500 votes [9, Chapter 7]. It was suspected that somebody else was using her ID card. When she was contacted, the voter confirmed revoting, and stated that it was perfectly legal. No other information is available to explain her behaviour.

2.7.13 Revoting using a different eID tool

If a repeated vote is cast using another eID tool it might be that another person is using the voter's credentials. However, it could also happen if the voter was simply testing her other eID tool for i-voting.

2.7.14 Revoting using the same eID tool but different certificates

According to the law a voter is allowed to have only a single pair of certificates for a single type of eID tool, and therefore we should not see voting sessions that use the same eID tool but different certificates. If that happens it might indicate an NCA compromise, where an attacker has fraudulently obtained an additional eID tool with the voter's identity.

However, it is also possible that the certificates of the voter's eID tool expired during the voting period and the voter is revoting with renewed certificates.

2.7.15 Revoting from different IP addresses

The voter might revote from a different location and thus a different IP address if in the previous voting session she was coerced (e.g., family voting, voting in the workplace). Revoting in a short time frame using IP addresses that are physically distant from each other (e.g., another country) should be suspicious.

2.7.16 Parallel voting sessions

We define a parallel voting session as a session where the candidate list is requested between a candidate list request and a vote submission request of another voting session of the same voter.

The IVCA does not forbid running several IVCA instances at the same time in the same computer. The i-voting protocol enforced on the server side also does not prevent the voter from having several parallel voting sessions in progress.

Parallel voting sessions that do not share the same IP address, eID tool and OS could indicate the race condition between the attacker and the legitimate voter.

2.7.17 Vote verified from different IP addresses

This could happen in a situation where the voter uses multiple mobile devices to verify her vote or restarts the network connection between verifications. If the vote is verified from more than a few IP addresses, it could indicate that vote verification QR code has somehow become available to third persons (e.g., by publishing it on the Internet).

[2] The revoter proportion provided here is the upper bound calculated from the number of replaced votes, assuming that the revoters revoted only once.

2.7.18 Voter's votes verified from different IP addresses

Since the voter's identity is not included in the verification protocol message returned to the verification application, an attacker who has compromised the voting device can change the vote and escape detection by replacing the QR code shown to the verifier with a QR code that contains a different candidate and is cast by a different voter [4, Section V.E]. The attacker could obtain such QR codes by revoting. Thus, a revoter whose votes are verified from different IPs could be such an attacker.

Alternatively, an attacker could use QR codes from voters who do not verify their votes[3]. This approach would not trigger this specific anomaly, but would require the attacker to compromise as a minimum two times more voting clients than the number of clients against whom the vote changing attack can be executed and would still risk being detected by the verifier if the malware fails to correctly identify the non-verifier or the non-verifier revotes soon after voting, thereby making the vote reference contained in the QR code unusable.

2.7.19 IP address shared by several verifiers

Such a common verification IP address could be the coercer's or the vote buyer's Internet connection used to verify that the vote has been cast for the expected candidate. This could also happen if one mobile device or Internet connection is shared by several verifiers or the IP address is reassigned to different mobile devices.

2.7.20 First voting session seen as revoting

Before a candidate list is shown to a voter, the IVCA displays a note which shows to the voter whether she is going to cast the vote for the first time in these elections or if the vote has already been cast. This allows the voter to detect if someone else has voted on behalf of the voter, and contact the NEC for investigation.

2.7.21 Non-i-voter denied paper vote

On the election Sunday voters who have i-voted have the "I" mark registered under their name in the voter list in their polling station and thus, they are not allowed to cast a paper vote. A paper voter who on election Sunday has come to vote would detect if someone has i-voted on behalf of her during the advance voting period and turn to election officials for investigation.

2.7.22 I-voting results deviating from paper voting results

A dishonest candidate running an attack on i-voting to increase her chances of being elected would likely have a disproportional i-voter and paper voter ratio that would be visible in the election results. However, a less selfish attacker working for the benefit of the whole political party could improve the party's results by distributing fraudulent votes to all the candidates of the political party and thus escape the attention. This is possible because differences in the i-voting and paper voting ratio between parties are common and can be plausibly explained by a party electorate having different support towards i-voting.

2.7.23 General statistics

General statistics about the age and gender of voters and verifiers, as well as OS and eID tool usage might reveal interesting i-voting characteristics. Significant differences in these characteristics between elections could be a sign of irregularities in the i-voting, which must be investigated and explained.

2.7.24 Other irregularities

This anomaly pattern covers all other sufficiently important irregularities related to i-voting that do not fall under other anomaly patterns described in this section.

[3] A malware could detect such voters with some certainty by measuring how fast the IVCA is closed after the verification QR code is shown.

3 Results from KOV2013

The i-voting in KOV2013 (Municipal Elections 2013 [10]) took place from 10 October 2013 at 09:00 to 16 October 2013 at 18:00.

4,086,492 log entries from 10 October 2013 at 09:00:23 to 16 October 2013 at 18:34:09 were analysed. The starting point of the analysis was the moment when the voting period actually started. The ending point of the analysis was the moment when no more votes were accepted.

The voting period was started and stopped by a human operator. The voting period is stopped gradually – at 18:00 the distribution of candidate lists is stopped and votes are accepted only from those voters who downloaded the candidate list before that time. The time of not accepting any more votes is decided by a human operator. It is similar to the situation in the polling station, where no new ballots are being issued, but voters are allowed to fill in the ballots they already have.

The breakdown of sessions, the number of unique voters connected to these sessions and the number of voters who did not manage to successfully i-vote (column "Voters (u)") is given in Table 1.

Session kind	Sessions	Voters	Voters (u)
All sessions	176,144	–	–
Voting	170,801	138,532	4,724
Successful	136,853	133,808	0
ID card	125,100	122,471	0
Mobile-ID	11,753	11,395	0
Unsuccessful	33,948	19,705	4,724
ID card	26,103	16,201	4,102
Mobile-ID	7,845	3,658	655
Verification	5,343	4,542	21
Successful	5,024	4,521	0
Unsuccessful	319	84	21

Table 1: KOV2013: Session breakdown

3.1 Unexpected log entries

In total 93 voting sessions raised an incident caused by unexpected log entries. Here we provide a grouped summary of them.

3.1.1 Communication problem with the VSS

On 15 October 2013 from 15:12:26 to 15:13:08, 36 failed voting sessions were logged with an incident message informing about the unavailability of the VSS. The reason for the VSS downtime was a vote backup routine that required stopping the Apache process running on the VSS. Starting from the next elections (EP2014), LVM snapshots will be used. This will allow to back up votes without stopping the Apache process.

3.1.2 Invalid vote

A total of 11 incidents were logged stating that the submitted vote was invalid since it did not contain a signature or certificate data. The error was traced down to a bug in the IVCA. The IVCA continued with vote submission even if the certificate could not be read from the smart card or if digital signature generation in the smart card failed. In total eight voters were affected – all of them were Windows OS users using an ID card to cast the vote. After retrying all of them managed to successfully cast the i-vote. This bug in the IVCA has been fixed.

3.1.3 Invalid digest of an ID card signing certificate

We observed 37 ID card voting sessions that failed with the incident message stating that the signing certificate digest did not match the digest specified in the BDOC (vote). In total 12 voters were affected. All of the voters were using Linux OS except one voter who was using Windows OS and was the only one voting from an IP address located outside Estonia – in Germany. All the voters except the voter from Germany and the voter who tried to i-vote on the last i-voting day were able to recast their vote successfully. The incident was traced to a bug in the smart card library OpenSC [11] shipped with some Linux distributions. The bug resulted in failure to remove zero padding from the certificate when reading it from the smart card.

3.1.4 Invalid ID card signature

We observed two ID card voting sessions that failed with the incident message stating that the signature of the vote was invalid. Both voting sessions where carried out by two voters using Windows OS and voting from the same IP address with an interval of just a few minutes. The voters were male and female, both born in 1965. Two days later both voters successfully revoted using two different IP addresses and with an interval of several hours between the voting sessions. Without the corresponding invalid votes, we were not able to investigate what caused this incident.

3.1.5 Invalid signature of an ID card signing certificate

We observed two ID card voting sessions that failed with the incident message stating that the certificate which was used to sign the vote had an invalid signature. The voting sessions were carried out on different i-voting days by different voters both using Windows OS. However, just a few minutes later, both voters were able to revote successfully. Without the corresponding invalid votes, we were not able to investigate what caused this incident.

3.1.6 Invalid Mobile-ID RSA signature

We observed two Mobile-ID voting sessions from a single voter that failed with the incident message stating that the Mobile-ID signature received from the NCA DigiDocService could not be verified (`RSA_public_decrypt()` failed). Four minutes later the voter successfully revoted using an ID card. This case was investigated by the NCA and it was found that the voter's SIM card was defective.

3.1.7 Invalid phone number

We observed three Mobile-ID voting sessions which raised an incident about an invalid phone number. The problem was traced down to the IVCA that failed to correctly enforce a valid phone number input from the voter. This bug in the IVCA has been fixed.

3.2 Incorrect session state change

No incidents caused by an incorrect session state change were observed.

3.3 Unsuccessful voting sessions

In the normality profile we have defined that a voting session should end with a successfully cast vote. In practice out of 170,801 voting sessions 33,948 (19.88%) sessions involving 19,705 voters did not result in a successfully cast vote.

The breakdown of error conditions, the number of unique voters affected in these voting sessions and the number of voters who did not manage to successfully i-vote (column "Voters (u)") is given in Table 2. The Table 3 further details issues specific to Mobile-ID.

Reason for failure	Sessions	Voters	Voters (u)
Unsuccessful voting sessions	33,948	19,705	4,724
Explicit error	8,979	4,207	1,954
Common error	1,103	811	793
Maintenance	11	11	1
Under-aged voter	28	25	25
Ineligible voter	1,063	774	766
Voting ended	1	1	1
Certificate issue	1,978	872	755
ID card	1,933	872	755
Mobile-ID	45	–	–
Pre-2011 Mobile-ID user	1,490	876	332
Bad Mobile-ID number	2,051	–	–
DigiDocService failure	47	28	2
Authentication	27	9	1
Signing	20	19	1
Mobile-ID failures	2,217	1,656	100
Incident	93	60	6
Other reason	24,969	16,087	2,965
Discontinued (Mobile-ID)	826	595	68
Authentication	636	470	62
Signing	190	178	10
Abnormal	40	34	30
Vote not submitted	24,103	15,563	2,889
ID card	23,004	14,630	2,689
Mobile-ID	1,099	954	202

Table 2: KOV2013: Unsuccessful voting sessions

Some of the unsuccessful voting sessions (8,979 sessions, 4,207 voters) failed with an explicit error condition. The 4,207 voters included 1,954 voters who did not manage to successfully i-vote.

Reason for failure	Sessions	Voters	Voters (u)
Mobile-ID failures	2,217	1,656	100
User cancelled	429	381	24
Authentication	261	239	19
Signing	168	166	7
Not in coverage	85	66	12
Authentication	69	52	12
Signing	16	16	0
SIM error	214	145	5
Authentication	138	89	4
Signing	76	75	1
SMS sending error	246	168	15
Authentication	194	130	13
Signing	52	48	2
Other	1,243	1,035	55
Authentication	702	613	44
Signing	541	497	14

Table 3: KOV2013: Mobile-ID failures

In the largest share of unsuccessful voting sessions (24,103 sessions, 15,563 voters), the candidate list was successfully downloaded, but a vote submission request did not follow. From these 15,563 voters 2,889 voters did not manage to cast an i-vote. From these 2,889 voters 176 had at least one voting session which failed. From the remaining 2,713 voters, 2,000 voters had carried out a single voting session that did not continue after candidate list retrieval; 370 voters had two such sessions, while 52 voters had more than six such sessions. We also observed nine voters who obtained the candidate list more than 15 times in a row and then cast their vote in their last voting session.

Some of these unfinished voting sessions can be explained by a bug [12] in the libcurl library used by the IVCA, which causes a connection timeout when sending a vote submission request over a slow network connection.

Some unsuccessful voting sessions were Mobile-ID sessions that were discontinued in the Mobile-ID authentication or signing phase. This could have been caused by a software error or a user closing the IVCA in the middle of the process.

We observed 40 voting sessions (involving 34 unique voters) which were in an abnormal state, i.e., we saw that the candidate request was made, but no further log entries about the fate of the candidate request followed. All sessions were initiated on the last i-voting day, which was 16 October 2013, from 18:01:59 to 18:34:09, after i-voting was terminated. This was traced down to a missing error logging in the server-side code in case the candidate list was requested after the candidate list issuance was terminated. This bug in the server-side code has been fixed.

In 2,120 cases it was not possible to identify the voter associated with the unsuccessful voting session. These cases were exclusively Mobile-ID voting sessions and the vast majority of those (2,051) were due to the fact that the phone number was not associated with the Mobile-ID capable SIM card.

From the 138,532 persons who attempted to i-vote in KOV2013, 133,808 (96.59%) managed to cast at least one succesful vote.

3.4 Unsuccessful verifications and verification sessions

From all the i-voters 4,542 (3.39%) attempted to verify their i-vote. In the KOV2013 elections the NEC received no complaints about unsuccessful vote verification.

However, we see that from 5,343 verification requests 319 (5.97%) were unsuccessful.

The breakdown of reasons, the number of unique verifiers affected in these unsuccessful verification sessions and the number of verifiers who did not manage to successfully verify any vote (column "Verifiers (u)") is given in Table 4.

Reason for failure	Sessions	Verifiers	Verifiers (u)
Unsuccessful sessions	319	84	21
Newer vote cast	19	6	0
Verification count exceeded	144	47	6
Verification time exceeded	95	54	21
Abnormal state	1	1	0
Vote ID not issued	60	–	–

Table 4: KOV2013: Unsuccessful verification sessions

Most of the verification failures were caused by voters trying to verify the same vote more than three times or after the time allowed for vote verification had passed.

If we look at a voter's first verification attempt, we see that for 33 voters their first verification attempt was not successful, resulting in an error message shown to the voter (31 – tried to verify after 30 minutes, 1 – after submitting a newer vote, 1 – when the VSS was unreachable).

It is interesting to note that ten voters made their first verification request six hours after the vote was submitted, and six voters even a day after. Most likely these verifiers faced problems when installing the verification application.

We believe that these voters did not contact the NEC because they suspected that the verification failure was caused by their verification peculiarities.

We did also observe one verification session made on 15 October 2013 at 15:12:54 in an abnormal state, i.e., we saw that the verification request was made, but no further log entries followed about the fate of this verification request. We found that this verification request was made at a time when the VSS was unavailable and that the VFS silently rejected the verification request without logging the error message. This bug in the server-side code has been fixed.

We observed a total of 60 vote verification requests for three unique vote identifiers that were not issued in the KOV2013 elections. First of them was queried one time, the second one was queried two times from a single IP address, but the third one was queried 57 times from 24 unique IPs. We suspect that these vote identifiers are of QR codes from test elections that have been published somewhere (e.g., in the NEC documentation) and curious people are using them to see to which candidates the corresponding votes were given.

3.5 Support requests handled by the NEC support centre

In the KOV2013 elections the NEC support centre registered 257 support requests. The breakdown by topics is shown in Table 5.

Android users on Android 2.3.6 (Samsung Galaxy Young, LG-E400) and 4.0.4 (Sony Xperia Acro S) reported a VVA crash after pressing the next-button.

Topic	#
QR code focussing problems	8
State-revoked ID cards (issued in 2011)	5
Android VVA crash	3
IVCA Internet connectivity issues	109
Unsupported voting platforms	3
Pre-2011 Mobile-ID user	6
IVCA bad server response error	3
ID-software not installed	13
PIN code issues	9
ID-software, card reader drivers	13
Other	85

Table 5: KOV2013: Support requests handled

IVCA connectivity errors were caused mainly by exceedingly strict firewall rules or security software that tried to intercept encrypted communications. A large share of these errors in KOV2013 can be attributed to a bug in the library used by the IVCA (see Section 3.3).

Other topics include suggestions on improving the instructions available on the web, questions related to elections in general, damaged/dirty ID card chips, the process of voting and verifying the vote, ID cards inserted upside down in the reader, updating certificates, etc.

3.6 Voting sessions too slow

In the normality profile we described that a voting session should be completed in a few minutes. Figure 3 shows the histogram of actual voting session lengths observed in KOV2013.

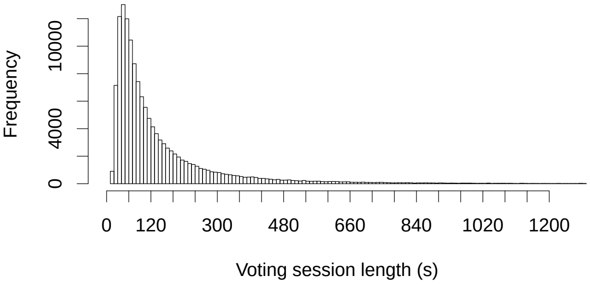


Figure 3: KOV2013: Distribution of voting session lengths

Minimal and maximal session lengths were 11 seconds and 408,200 seconds (about 4.72 days), respectively. The mean length was 171.7 seconds and median length 88 seconds. 0.5%, 1%, 99% and 99.5% quantiles are given in Table 6.

Quantile	0.5%	1%	99%	99.5%
Value (s)	20	22	1,182	1,685.4

Table 6: KOV2013: Quantiles of voting session lengths

The table allows us to estimate that the normal length for a voting session could be between 20 seconds and 20 minutes. Note that for 91.28% of the voting sessions the session was shorter than six minutes. Note that there were 1,364 sessions that were longer than 20 minutes, 140 sessions that were longer than an hour, 45 sessions that were longer than two hours, 15 sessions that were longer than six hours, 5 sessions that were longer than a day.

The slowest voting session, which took 4.72 days, was carried out by a male born in 1976, who used Mac OS and an ID card to cast the vote. No other activity was observed from the same person.

The three fastest voting sessions (11, 12 and 12 seconds) were made by an NEC employee, who was performing healthiness tests of the i-voting system.

3.7 Vote signed with a different eID tool

No such sessions were observed.

3.8 IP address or OS change in the middle of a voting session

We observed 72 voting sessions affecting 72 unique voters where the vote submission IP address was different from the candidate list retrieval IP address. The sessions were timewise evenly distributed over the i-voting period and the OS in these sessions did not change. In 61 sessions the IP changed from one Estonian IP to another IP in Estonia. In ten sessions the IP changed from one foreign IP to another IP in the same country.

In one voting session the IP changed from one country to another – the candidate list was requested from an IP address in the US which was assigned to a VPS hosting provider, but on vote submission the IP address (after 4 minutes and 15 seconds) changed to an IP address in China, and this IP address had a track record in projecthoneypot.org of being used for spam and dictionary attacks. The voting session was carried out by a male born in 1983, who was using Mac OS and an ID card to cast the vote. No other activity from the same person and IP addresses were observed.

3.9 IP address shared by several voters

In the KOV2013 elections, 133,808 voters used 68,503 unique IP addresses to cast their successful votes, which means that in KOV2013 on average 1.95 persons shared one IP address.

There were 28 IP addresses that were each shared by more than 100 voters with the top IP shared by 1,127 voters. We reviewed the top shared voting IPs and did not notice any strange patterns – voting was evenly distributed over the voting period, different OS versions were used and several voting sessions overlapped.

We observed a large number of IP addresses shared by two and more voters where the voting sessions were not evenly distributed over the voting period, with the voters casting their votes shortly after each other. Table 7 shows the number of voter groups observed, where voters voting in five-minute intervals and using the same OS are considered as one group. The table contains data only on these IP addresses that do no have overlapping voting sessions and these whose first and last voting activity falls in an 24-hour window.

Voters	Groups
2	8,476
3	697
4	108
5	15
6	3

Table 7: KOV2013: Voter groups

3.10 Non-unique vote encryption

All of the votes received had a unique ciphertext.

3.11 Large percentage of revoters

From 133,808 voters 2,586 (1.93%) voters cast more than one vote. From these revoters 2,359 voted two times, 186 voted three times, and 41 voted four or more times.

The distribution of time between the revoters' first and second vote is shown in Figure 4. We can see that 30% of revoters revote in the first ten minutes, and 41% of revoters revote in the first hour after casting their vote.

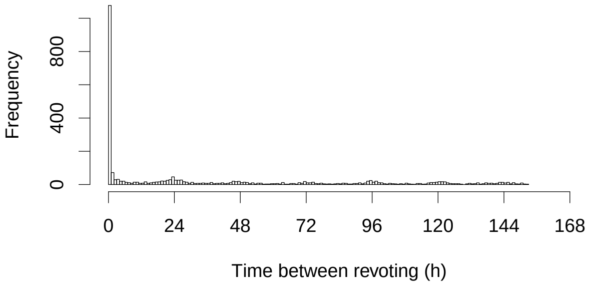


Figure 4: KOV2013: Distribution of time between revotes

Figure 5 shows the distribution of votes and revotes over the voting period. We see that revotes are evenly distributed over the voting period.

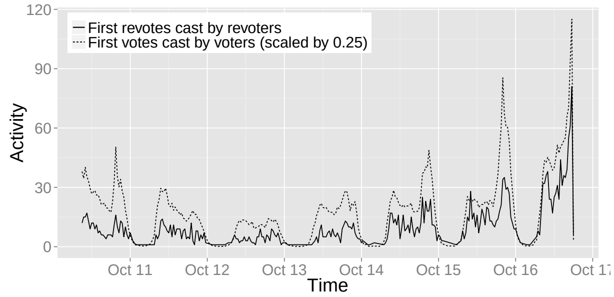


Figure 5: KOV2013: Distribution of votes and revotes

We can estimate that in the worst case, 2,586 voters in the KOV2013 elections could have been coerced or fallen victim to the revoting malware described in Section 2.7.11.

However, since in the previous elections the revoter proportion was similar (see Section 2.7.11) and some amount of revoters is normal, it is unlikely that most of the revotes were caused by an attack.

3.12 Voters revoting many times

The top ten revoters cast 41, 39, 36, 28, 20 17, 11, 9, 7, 7 number of votes. Revoters who cast 39 and 17 votes were identified as NEC employees who were testing and demonstrating the i-voting system.

3.13 Revoting using a different eID tool

In total 62 (2.47%) revoters used more than one eID tool to cast their vote. In 34 of these cases the voter also used a different IP address to revote.

3.14 Revoting using the same eID tool but different certificates

This check was implemented only after KOV2013 took place.

3.15 Revoting from different IP addresses

In total 539 (20.84%) revoters revoted from a different IP address. From these 539 voters 18 revoted from an IP in a different country. In all except one case, the same eID tool was used to revote and the time difference between revotes was large enough for the voter to physically change his country of location.

3.16 Parallel voting sessions

We observed 60 voters who had parallel voting sessions. In all cases the voting session was carried out using the same eID tool from the same IP address and using the same OS. These parallel voting sessions were most likely made from the same computer. Why these voters opened parallel IVCA instances is unclear.

3.17 Vote verified from different IP addresses

There were 18 votes which were verified from two different IP addresses and one vote which was verified from three different IP addresses. Since these vote identifiers were verified only from a few IP addresses, we can conclude that in KOV2013 no QR code was made available to the general public.

3.18 Voter's votes verified from different IP addresses

There were 67 voters whose votes were verified from more than one IP. A summary of these voters aggregated by the number of different verification IPs can be seen in Table 8.

Verification IPs	2	3	4	5	7
Voters	61	3	1	1	1
Max (votes)	7	5	3	39	36

Table 8: KOV2013: Summary of voters whose votes were verified from different IP addresses

The top two revoters – one whose votes were verified by seven different IPs and one[4] whose votes were verified by five different IPs – could be the attackers described in Section 2.7.18.

Thus we can give a weak estimate that altogether 12 verifiers might have been successfully attacked without detection using the attack described in [4, Section V.E].

From this estimate we excluded other revoters because of the negligible number of votes cast by these voters. The attacker who would want to successfully attack three verifiers without being detected whould need at least ten times more revotes since less than 4% of voters verify their vote.

3.19 IP address shared by several verifiers

There were 746 IP addresses which were each shared by several verifiers. The top IP address registered to the mobile operator EMT was shared by 62 verifiers. The remaining IP addresses were each shared by ten or fewer verifiers.

In KOV2013 4,542 verifiers used 3,364 unique IP addresses to verify their votes, which means that in KOV2013 on average 1.35 persons shared one verification IP address.

We see that in KOV2013 53.28% of verifiers verified their vote from the same IP address that was used to cast the vote.

3.20 First voting session seen as revoting

No cases have been registered by the NEC.

3.21 Non-i-voter denied paper vote

No cases have been registered by the NEC.

3.22 I-voting results deviating from paper voting results

In KOV013 21.2% [2] of the votes where i-votes. Thus, on average, a candidate received 21.2% i-votes. Table 9 shows candidates who have received the highest proportion of i-votes and who have received at least 30 i-votes. We see that the proportions and number of i-votes received are too small to raise suspicion.

Candidate	p-votes	i-votes	i-votes (%)
OLLE KOOP	11	33	75.00%
ALLAN ALLMERE	16	37	69.81%
MOONIKA ORAS	18	40	68.96%
JAAK RAIE	22	44	66.66%
KATI KÄPP	19	38	66.66%
KAIRI UUSTULND	45	87	65.90%
KARIN PÄRTEL	20	38	65.51%
ELVER LOHO	21	38	64.40%
SERGEI ŽUKOV	18	32	64.00%
KÄTLIN VAU	21	35	62.50%

Table 9: KOV2013: Candidates with the ten highest i-vote proportions

[4] This revoter has been identified as a NEC employee.

3.23 General statistics

3.23.1 Age distribution

The youngest person who (unsuccessfully) attempted i-voting was three years old, and the oldest i-voter was 102. The youngest vote verifier was 18 and the oldest was 97. The voter turnout by age is shown in Figure 6. We see that the most active voters are people aged 35.

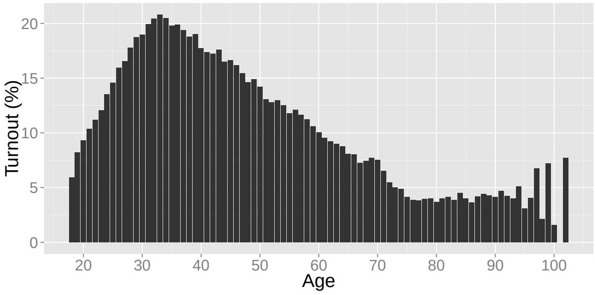


Figure 6: KOV2013: Voter activity by age

The percentage of voters by age who verified their vote is shown in Figure 7. We see that verifier activity by age is more uniform than voter activity.

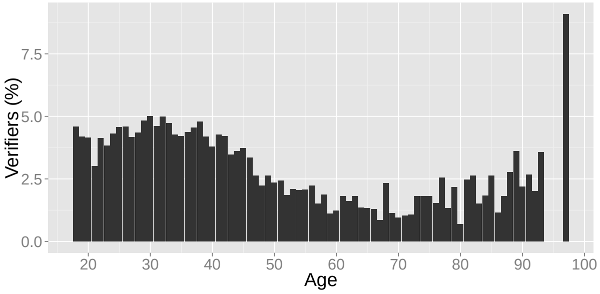


Figure 7: KOV2013: Verifier activity by age

An interesting phenomenon was observed when studying the relationship between the voting session length and the voter's age. It turns out that older people are faster voters. The phenomenon is illustrated in Figure 8 (taking into account only the voters' first voting session and excluding sessions longer than 20 minutes). We note that this phenomenon does not disappear when splitting the data by gender or the eID tool used. This can be explained by the fact that multitasking is less popular among older people.

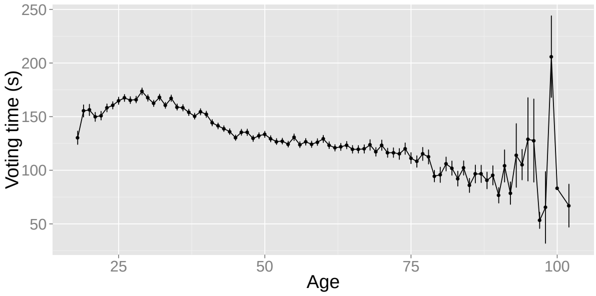


Figure 8: KOV2013: Age vs voting time

3.23.2 Gender distribution

It has been observed for the last elections that more votes are cast by females. KOV2013 was no exception. Out of all the successful voters 52.2% were females.

However, if looking at the turnout, we see that 12.94% of eligible males i-voted, while out of all eligible females 11.78% i-voted. Thus we see that in KOV2013 males were 1.16% more active than females.

Figure 9 shows the percentage of female voter activity by age. We see that male and female activity is quite uniform.

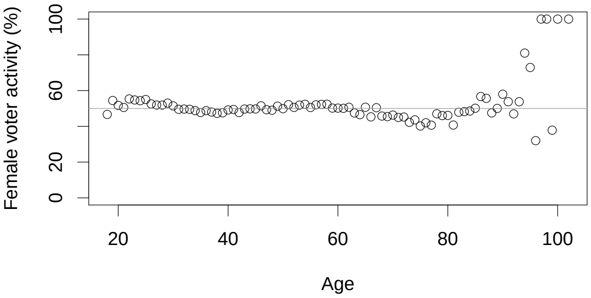


Figure 9: KOV2013: Female voter activity by age

From all 4,542 verifiers 1,424 (31.6%) were female. We see that 4.87% of male voters and 2.04% of female voters verified their vote. Thus, in KOV2013, male voters were 2.38 times more active as verifiers than female voters. Figure 10 shows the percentage of female verifier activity by age.

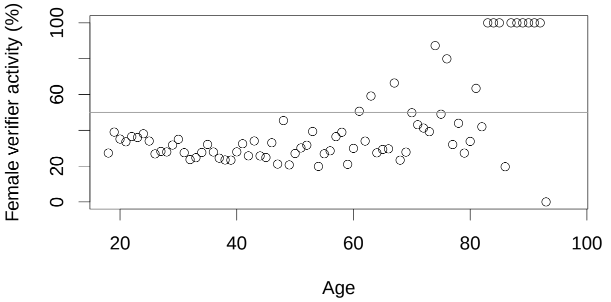


Figure 10: KOV2013: Female verifier activity by age

3.23.3 OS distribution

The official IVCA is available for three OSs. From all the successfully cast votes (excluding votes annulled by revoting) the most popular OS was Windows at 93.87%, then Mac at 5.35%, and finally Linux at 0.78%. OS distribution by age is shown in Figure 11. OS distribution by gender is shown in Figure 12.

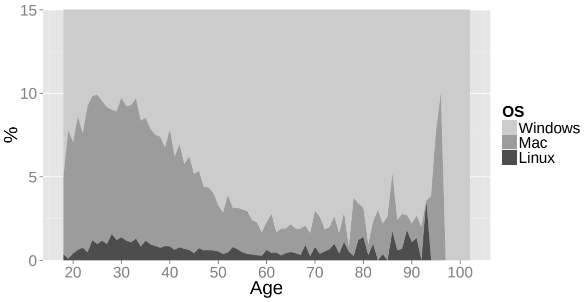


Figure 11: KOV2013: OS distribution by age

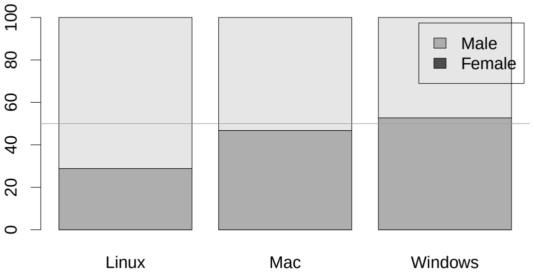


Figure 12: KOV2013: OS distribution by gender

3.23.4 eID tool

An i-vote can be cast using three eID tools. From all the successfully cast votes (excluding votes annulled by revoting) the most popular eID tool was the ID card at 90.27%, then Mobile-ID at 8.49%, and finally Digi-ID at 1.23%. eID distribution by age is shown in Figure 13. There we can see that Mobile-ID is especially popular among 30-year-olds. eID distribution by gender is shown in Figure 14.

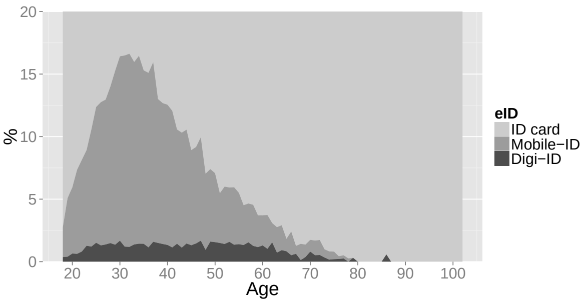


Figure 13: KOV2013: eID distribution by age

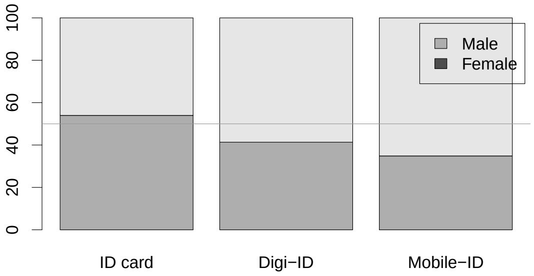


Figure 14: KOV2013: eID distribution by gender

3.23.5 Verification

From all 4,542 verifiers 413 cast more than one i-vote, while 82 (19.85%) verified all their i-votes. From the 331 verifiers who did not verify all their i-votes, 282 (85.2%) verified their last vote.

We see that Mobile-ID holders (voters who cast at least one vote using Mobile-ID) are 3.76 times more active verifiers than non-holders, since 10.34% of Mobile-ID holders verified their vote, while only 2.75% non-holders verified their vote.

The time between the moment when the vote identifier is issued and the vote verification request is received is called "verification length". A voter can verify the same vote a maximum of three times but only within 30 minutes after submitting vote and before the voter has submitted a new i-vote. The frequencies of verification lengths (taking into account only the first verification request made by the voter) are shown in Figure 15.

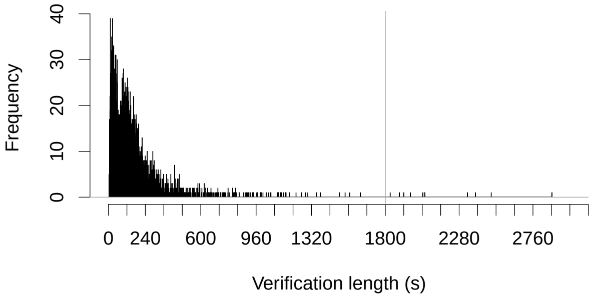


Figure 15: KOV2013: Distribution of verification lengths

Table 10 shows how many times (at least) the voters verified their first vote and the corresponding success rate for consecutive verifications. We see that most voters do not perform more than one verification.

Times	1	2	3	4	5	6	7	8	9	10
Voters	4,542	333	68	42	18	15	12	10	7	6
Success	99.27%	90.12%	61.76%	–	–	–	–	–	–	–

Table 10: KOV2013: Distribution of verification counts

3.24 Other irregularities

3.24.1 Invalid vote cast

Similarly as in RK2011 [3, Section 3.1] it was found in the vote tallying process that the encryption of one vote is invalid. Some source has shared a link to pastebin which contained instructions on how to use the GNU debugger to locate a breakpoint in the Linux IVCA where the encrypted vote is stored. We suspect that this kind of manipulation of the IVCA has been used to cast this invalid vote.

4 Results from EP2014

The i-voting in EP2014 (the 2014 European Parliament Elections [13]) took place from 15 May 2014 at 09:00 to 21 May 2014 at 18:00.

3,024,107 log entries from 15 May 2014 at 09:00:21 to 21 May 2014 at 18:22:07 were analysed. The starting point of the analysis was the moment when the voting period actually started. The ending point of the analysis was the moment when no more votes were accepted.

The breakdown of sessions, the number of unique voters connected to these sessions and the number of voters who did not manage to successfully i-vote (column "Voters (u)") is given in Table 11.

Session kind	Sessions	Voters	Voters (u)
All sessions	120,503	–	–
Voting	114,792	104,679	1,528
Successful	105,157	103,151	0
ID card	93,558	91,964	0
Mobile-ID	11,609	11,226	0
Unsuccessful	9,625	6,050	1,528
ID card	6,248	4,157	1,218
Mobile-ID	3,377	1,940	318
Verification	5,711	4,250	40
Successful	4,894	4,210	0
Unsuccessful	817	131	40

Table 11: EP2014: Session breakdown

4.1 Unexpected log entries

In total 1,173 voting and 196 verification sessions raised an incident caused by unexpected log entries. We also observed two ID card and four Mobile-ID voting sessions with incorrect state transitions. Additionally, five ID card voting sessions with inconsistent data were observed.

Here we provide a grouped summary of them.

4.1.1 Malformed vote verification requests

We observed 196 vote verification requests received from 38 unique IP addresses having a malformed vote ID. The malformed vote verification requests were traced back to the iOS-based vote verification application, which failed to validate the contents of a captured QR code before forming a vote verification request sent to the VFS. This bug in the iOS-based vote verification application has been fixed and the server-side code has been fixed to provide more verbose error logging.

4.1.2 Invalid BDOC `signatures0.xml`

We observed 41 Linux IVCA voting sessions using the ID card eID tool, which failed with an error message stating that the `signatures0.xml` in the submitted BDOC vote is too large. A total of 14 voters were affected, from whom, 12 voters later successfully recast their i-vote. The problem was attributed to an OpenSC certificate padding bug [11] also observed in KOV2013 (see Section 3.1.3). The fixed OpenSC version has been shipped with most Linux distributions and therefore this bug will most likely not be encountered in RK2015.

4.1.3 Certificate not yet valid

We observed 1,131 voting sessions that failed with an error message stating that the certificate used to sign the vote was not yet valid. The error was traced back to a bug in the server-side software introduced in EP2014, which did not take into account timezone information when checking the validity date. The error affected voters who had renewed their eID tool on the day of i-voting. Voters who approached the NEC support centre were instructed to retry after a few hours. From the 310 voters affected 229 managed to successfully cast their i-vote later in the i-voting period.

4.1.4 Invalid ID card RSA signature

We observed one ID card voting session using Windows IVCA that failed with the incident stating that the vote signature could not be verified (`RSA_public_decrypt()` failed). Three minutes later the voter successfully revoted using the same ID card authentication certificate, but a different digital signature certificate. The hash of the digital signature certificate used in the failed voting session could not be found in any other voting session. We suspect that the voter swapped her currently valid ID card before signing the vote with her older ID card which had been officially reported as lost.

4.1.5 Vote submitted by a different person

We observed five ID card voting sessions where the person submitting the vote was not the same person who obtained the candidate list. This behaviour can be explained by the new "Retry" button feature introduced in the IVCA which allows a person to obtain the candidate list using one smart card, but sign and submit their vote with another by swapping cards between these operations. These votes were accepted and counted without a problem. While it is not an issue in the European Parliament elections, it may happen that the voter obtaining the candidate list and the voter casting the vote has different candidate lists, which will result in an invalid vote in the vote counting phase. Therefore, server-side software was modified to reject the vote if the candidate list was not obtained by the same person who cast the vote.

It is not clear why these five voters decided to swap their ID card with another person's ID card before signing the vote. The persons involved in these sessions were elderly couples (M79 and F72, M50 and F71, F68 and M74, M56 and F58, F33 and M52). From the voters who obtained the candidate list two gave their own vote few minutes later; however, three voters did not cast their own vote.

4.2 Incorrect session state change

We observed two ID card and four Mobile-ID voting sessions where more than one vote was submitted in a single voting session triggering an incident about an illogical session state change. The behaviour was caused by an introduced feature in the IVCA, which allows the voter to resubmit the vote if vote submission has failed. Most likely, in these sessions the first vote submission response timed out and the response about successful vote submission did not reach the IVCA, allowing the voter to submit the vote again. Network timeouts are inevitable and therefore we must accept a certain number of such incidents.

4.3 Unsuccessful voting sessions

In the normality profile we have defined that a voting session should end with a successfully cast vote. In practice out of 114,792 voting sessions 9,625 (8.38%) voting sessions (19.88% in KOV2013) involving 6,050 voters did not result in a successfully cast vote.

The breakdown of error conditions, the number of unique voters affected in these voting sessions and the number of voters who did not manage to successfully i-vote (column "Voters (u)") is given in Table 12. The Table 13 further details issues specific to Mobile-ID.

Reason for failure	Sessions	Voters	Voters (u)
Unsuccessful voting sessions	9,625	6,050	1,528
Explicit error	4,032	1,920	654
Common error	369	249	242
Maintenance	0	0	0
Under-aged voter	16	16	15
Ineligible voter	315	199	199
Voting ended	1	1	0
No new voters	37	34	28
Certificate issue	302	146	128
ID card	270	146	128
Mobile-ID	32	–	–
Pre-2011 Mobile-ID user	549	407	160
Bad Mobile-ID number	491	–	–
DigiDocService failure	0	0	0
Mobile-ID failures	1,148	831	47
Incident	1,173	325	88
Other reason	5,593	4,340	914
Discontinued (Mobile-ID)	672	477	49
Authentication	461	332	31
Signing	211	196	19
Abnormal	0	0	0
Vote not submitted	4,921	3,889	869
ID card	4,524	3,521	797
Mobile-ID	397	371	72

Table 12: EP2014: Failed voting sessions

Some unsuccessful voting sessions (4,032 sessions, 1,920 voters) failed with an explicit error condition. From the 1,920 voters involved 654 voters did not manage to successfully i-vote.

In the largest portion of unsuccessful voting sessions (4,921 sessions, 3,889 voters) the candidate list was successfully downloaded, but the vote submission request did not follow. From these 3,889 voters 869 voters did not manage to cast their i-vote. From these 869 voters 20 voters had at least one voting session that failed. From the remaining 849 voters, 700 voters (2,000 in KOV2013) had carried out a single voting session that did not continue after candidate list retrieval, 79 voters had two such sessions, nine voters had more than six such sessions. We can only guess why these voters did not get past candidate list retrieval. We could guess that these 700 voters with one voting session forgot their PIN2 or lost interest in voting once they saw the

Reason for failure	Sessions	Voters	Voters (u)
Mobile-ID failures	1,148	831	47
User cancelled	338	308	12
Authentication	176	163	6
Signing	162	157	6
Not in coverage	60	50	4
Authentication	54	45	3
Signing	6	6	1
SIM error	137	118	1
Authentication	66	58	1
Signing	71	70	1
SMS sending error	421	238	18
Authentication	358	191	16
Signing	63	63	2
Other	192	160	19
Authentication	96	79	10
Signing	96	91	11

Table 13: EP2014: Mobile-ID failures

candidate list. It is also possible that they did not understand that their selected choice had to be confirmed to be sent to the voting server. We also observed three voters (nine in KOV2013) who obtained the candidate list more than 15 times in a row and then cast their vote in the last voting session. One possible explanation could be that they were hoping to rotate in a different voting district and see a different candidate list; however, the EP2014 elections had the same candidate list in all voting districts, and thus this explanation is not plausible.

Some unsuccessful voting sessions were Mobile-ID sessions that were discontinued in the Mobile-ID authentication or signing phase. This could have been caused by a software error or a user closing the IVCA in the middle of the process.

There were no abnormal session interruptions in EP2014.

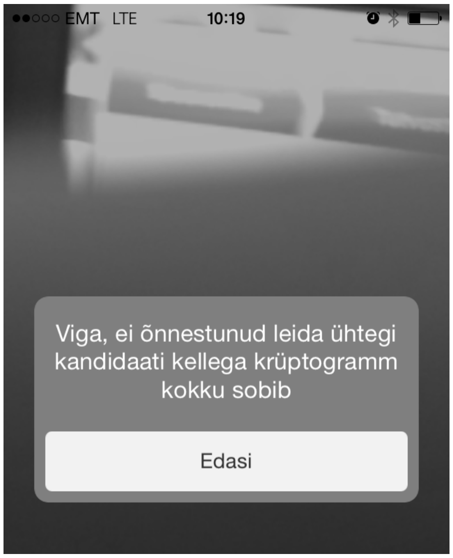
In 526 cases it was not possible to identify the voter associated with the unsuccessful voting session. These cases were exclusively Mobile-ID voting sessions and the vast majority of those (491) were caused by the fact that the phone number was not associated with the Mobile-ID capable SIM card.

From the 104,679 persons who attempted to i-vote in EP2014, 103,151 (98.54%) (96.59% in KOV2013) managed to cast at least one succesful vote.

4.4 Unsuccessful verifications and verification sessions

Out of all voters 4,250 (4.12%) (3.39% in KOV2013) attempted to verify their i-vote.

On the first day of i-voting in EP2014 the NEC received four complaints from iOS-based vote verification application users (all males) who complained about the error message “Error, failed to find a candidate who matches the cryptogram” displayed at vote verification (see Figure 16).



Viga, ei õnnestunud leida ühtegi kandidaati kellega krüptogramm kokku sobib

Figure 16: EP2014: Error in verification caused by a faulty iOS-based VVA

The error was traced to a bug in the iOS-based vote verification application that resulted in the incorrect handling of QR codes that contained a 0-byte in the RSA-OAEP padding. The bug was fixed and an updated iOS application was put in the iOS App Store on the second day of i-voting. During the i-voting period 559 voters (431 males and 128 females) tried to verify their vote using the faulty version of the iOS-based verification application. The probability of a randomly generated 20-byte RSA-OAEP padding to contain a 0-byte is 7.53%. Thus, we can estimate that around 42 voters were affected by this bug.

In the logs we see that from 5,711 verification requests 787 (13.78%) (5.97% in KOV2013) were unsuccessful.

The breakdown of reasons, the number of unique verifiers affected in these unsuccessful verification sessions and the number of verifiers who did not manage to successfully verify a vote (column "Verifiers (u)") is given in Table 14.

Reason for failure	Sessions	Verifiers	Verifiers (u)
Unsuccessful sessions	787	106	18
Newer vote cast	11	6	1
Verification count exceeded	317	81	5
Verification time exceeded	78	39	17
Malformed Vote ID	196	–	–
Vote ID not issued	185	–	–

Table 14: EP2014: Unsuccessful verification sessions

Most verification failures were caused by voters trying to verify the same vote more than three times or after the time allowed for vote verification had passed.

If we look at a voter's first verification attempt, we see that for 26 voters (33 in KOV2013) their first verification attempt was not successful, resulting in an error message shown to the voter (24 – tried to verify after 60 minutes, 2 – after submitting a newer vote).

It is interesting to note that ten voters (also ten in KOV2013) made their first verification request six hours after submitting the vote, and six voters (also six in KOV2013) even a day after. Most likely these verifiers faced problems when installing the verification application.

We observed a total of 185 vote verification requests for five unique vote identifiers (three in KOV2013) that were not issued in the EP2014 elections and also not in KOV2013. Three of these vote identifiers were queried by a single, distinct IP addresses. The fourth vote identifier was queried by 87 unique IPs. The fifth vote identifier was queried by 15 unique IPs and the same vote identifier was observed also in the KOV2013 analysis where it was queried by 24 unique IPs (see Section 3.4). These requests are again most likely made by curious people trying to verify the QR codes from the test elections.

4.5 Support requests handled by the NEC support centre

In EP2014 the NEC support centre registered 169 support requests. The breakdown by topics is shown in Table 15.

Topic	#
Android VVA crash	1
State-revoked ID cards (issued in 2011)	1
Pre-2011 Mobile-ID user	2
ID-software, card reader drivers	6
PIN code issues	9
Mac OS X without ID-software	41
QR code focussing problems	8
Website-related	14
Certificates not yet valid bug	10
iOS-based VVA 0-byte bug	4
IVCA Internet connectivity issues	24
Other	49

Table 15: EP2014: Support requests handled

An Android user on Android 4.0.3 (HTC One V) reported a VVA crash due to poor Internet connection.

The certificates not yet valid bug is described in more detail in Section 4.1.3. The iOS-based VVA 0-byte bug is described in more detail in Section 4.4.

IVCA connectivity errors were caused mainly due to excessively strict firewall rules or security software that tried to intercept encrypted communications.

Other topics included questions on if it was possible to vote using a smart phone, if it was possible to vote using a banklink, if a virtual keyboard was supported, if it was possible to vote without an ID card reader. There were questions concerning advance voting in polling stations, situations where the voter could not find the IVCA after downloading it, and submissions where the person had a problem but did not provide enough information to diagnose it.

4.6 Voting sessions too slow

In the normality profile we described that a voting session should be completed in a few minutes. Figure 17 shows the histogram of actual voting session lengths observed in EP2014.

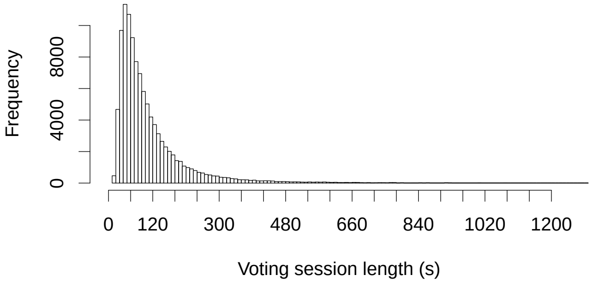


Figure 17: EP2014: Distribution of voting session lengths

Minimal and maximal session lengths were 11 seconds and 483,600 (about 5.6 days), respectively. The mean length was 140.3 seconds and median length 80 seconds. 0.5%, 1%, 99% and 99.5% quantiles are given in Table 16.

Quantile	0.5%	1%	99%	99.5%
Value (s)	21	23	751	1,080

Table 16: EP2014: Quantiles of voting session lengths

The table allows us to estimate that the normal length for a voting session could be between 20 seconds and 13 minutes (20 minutes in KOV2013). Note that for 96.11% voting sessions the session was shorter than six minutes. Note that there were 439 sessions that were longer than 20 minutes, 62 sessions that were longer than an hour, 32 sessions that were longer than two hours, 12 sessions that were longer than six hours, six sessions that were longer than a day.

The slowest voting session which took 5.6 days was made by a female born in 1991, who used Windows OS and an ID card to cast the vote. No other activity was observed from the same person.

The three fastest voting sessions (11, 11 and 11 seconds) were made by a male born in 1977 who was using Digi-ID to cast all of his 32 votes.

4.7 Vote signed with a different eID tool

There were no such sessions observed.

4.8 IP address or OS change in the middle of a voting session

We observed 46 voting sessions affecting 46 unique voters (72 in KOV2013) where the vote submission IP address was different from the candidate list retrieval IP address. The sessions were timewise evenly distributed over the i-voting period and the OS in these sessions did not change. In 30 sessions the IP changed from one Estonian IP to another IP in Estonia. In eight sessions the IP changed from one foreign IP to another IP in the same country and ISP. In four voting sessions the IP changed from one country to another. In the first case the voter obtained the candidate list from an IP address in Estonia registered to the mobile operator EMT, but seven minutes later submitted the vote from an IP in Spain registered to Vodafone Spain. In the second case the voter obtained the candidate list from an IP in Sweden, but four minutes later submitted the vote from IP in Estonia. In the third case the voter obtained the candidate list from an IP in Estonia, but two minutes later submitted the vote from an IP in Spain. In the fourth case the voter obtained the candidate list from an IP in Great Britain, but a minute later submitted the vote from an IP in the US.

4.9 IP address shared by several voters

In the EP2014 elections 103,151 voters used 52,191 unique IP addresses to cast their successful votes, which means that in EP2014 on average 1.97 persons (1.95 in KOV2013) shared one IP address.

There were 22 IP addresses (28 in KOV2013) that were each shared by more than 100 voters with the top IP shared by 970 voters (1,127 in KOV2013). We reviewed the top shared voting IPs and did not notice any strange patterns – voting was evenly distributed over the voting period, different OS versions were used and several voting sessions overlapped.

We observed a large number of IP addresses shared by two and more voters where the voting sessions were not evenly distributed over the voting period, with the voters casting their votes shortly after each other.

Table 17 shows the number of voter groups observed, where the voters voting in five-minute intervals and using the same OS are considered as one group. The table contains data only on these IP addresses that do no have overlapping voting sessions and these whose first and last voting activity falls in a 24-hour window.

Voters	Groups
2	6,033
3	523
4	60
5	9
6	1

Table 17: EP2014: Voter groups

4.10 Non-unique vote encryption

All the votes received had a unique ciphertext.

4.11 Large percentage of revoters

From 103,151 voters 1,743 (1.69%) voters (1.93% in KOV2013) cast more than one vote. From these revoters 1,600 voted two times, 100 voted three times, and 43 voted four times or more.

The distribution of time between the revoters' first and second vote is shown in Figure 18. We can see that 28% (30% in KOV2013) of revoters revote in the first ten minutes, and 38% (41% in KOV2013) of revoters revote in the first hour after casting their vote.

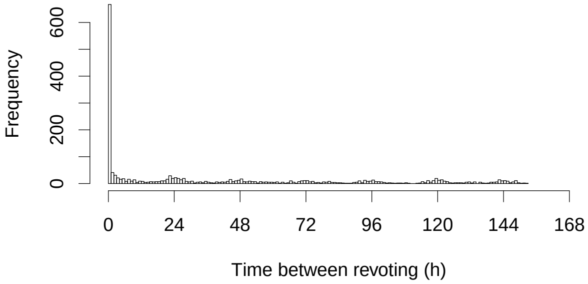


Figure 18: EP2014: Distribution of time between revotes

Figure 19 shows the distribution of votes and revotes over the voting period. We see that revotes are evenly distributed over the voting period.

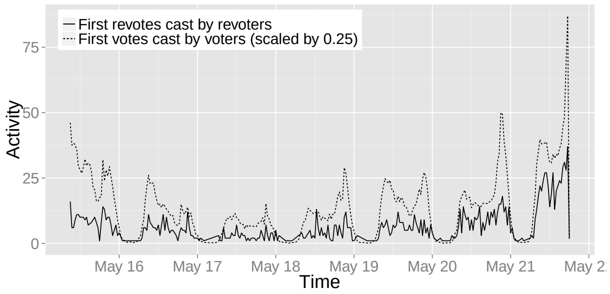


Figure 19: EP2014: Distribution of votes and revotes

We can estimate that in the worst case in the EP2014 elections 1,743 votes (2,586 in KOV2013) could have been replaced by a revoting malware described in Section 2.7.11.

However, since in the previous elections the revoter proportion was similar (see Section 2.7.11 and 3.11) and some amount of revoters is normal, it is unlikely that most of the revotes would have been caused by an attack.

4.12 Voters revoting many times

The top ten revoters cast 32, 27, 10, 10, 9, 8, 8, 7, 6, 6 votes. Revoters who cast 27, 10 and 9 votes were identified as NEC employees who were testing and demonstrating the i-voting system.

4.13 Revoting using a different eID tool

In total 41 revoters used more than one eID tool to cast their vote. In 23 of these cases the voter also used a different IP address to revote.

4.14 Revoting using the same eID tool but different certificates

We observed 22 voters who had at least two voting sessions using the same type of eID tool, but with eID tool containing different certificates. In case of 14 voters we see that the first voting sessions fail with a revoked certificate error and finally the vote is successfully cast using a different certificate (or fails because of the timezone bug described in

Section 4.1.3). In the remaining eight cases voting sessions using different certificates were carried out using the same IP address, except for one case where the IP address changed. In one case the voter's session raised an incident described in Section 4.1.4.

Thus we can conclude that certificate differences in these cases were not caused by an NCA compromise.

4.15 Revoting from different IP addresses

In total 358 voters (20.54%) (20.84% in KOV2013) revoted from a different IP address. From these 358 voters 26 revoted from an IP in a different country. In all cases the same eID tool was used to revote and the time difference between revotes was large enough for the voter to physically change his country of location.

4.16 Parallel voting sessions

We observed 28 voters (60 in KOV2013) who had parallel voting sessions. In all cases the voting session was carried out using the same eID tool from the same IP address and with the same OS. These parallel voting sessions were most likely carried out from the same computer. Why these voters opened parallel IVCA instances is unclear.

4.17 Vote verified from different IP addresses

There were 23 votes (19 in KOV2013) that were verified from more than one IP address – two different IP addresses. Since none of the vote identifiers were verified from more than two IP addresses, we can conclude that in EP2014 no QR code was made available to the general public.

4.18 Voter's votes verified from different IP addresses

There were 63 voters (67 in KOV2013) whose votes were verified from more than one IP. A summary of these voters aggregated by the number of different verification IPs can be seen in Table 18.

Verification IPs	2	3	4	5
Voters	55	5	2	1
Max (votes)	27	5	10	6

Table 18: EP2014: Summary of voters whose votes were verified from different IP addresses

From these revoters, two display verification activity that could indicate an attack described in Section 2.7.18. One who revoted 27 times and whose votes were verified from two different IPs and one revoted ten times and whose votes were verified from four different IPs. However, these two are unlikely to be attackers, since the verification IPs used belong to the same network segment. Since the attacker does not know from which IP the QR code will be verified, the IPs seen in the attack would be uniformly distributed among network segments.

We can also exclude other revoters due to the negligible number of votes cast by these voters. An attacker who would want to successfully attack three verifiers without being detected whould need at least ten times more revotes since less than 5% of voters verify their vote.

4.19 IP address shared by several verifiers

There were 721 IP addresses (746 in KOV2013) that were each shared by several verifiers. The IP addresses were each shared by 13 (ten in KOV2013) and fewer verifiers.

In the EP2014 elections 4,250 verifiers used 3,234 unique IP addresses to verify their votes, which means that in EP2014 on average 1.31 (1.35 in KOV2013) persons shared one verification IP address.

We see that in the EP2014 elections 56.82% (53.28% in KOV2013) of verifiers verified their vote from the same IP address that was used to cast the vote.

4.20 First voting session seen as revoting

No cases have been registered by the NEC.

4.21 Non-i-voter denied paper vote

No cases have been registered by the NEC.

4.22 I-voting results deviating from paper voting results

In the EP2014 elections 31.3% [2] of the votes where i-votes. Thus, on average a candidate received 31.3% i-votes. However, we see in Table 19, that some voters received as much as 61.75% i-votes. The proportion is close to other candidates and is not extreme enough to raise suspicion.

Candidate	p-votes	i-votes	i-votes (%)
ARTO AAS	135	218	61.75%
JÜRGEN LIGI	253	375	59.71%
KRISTA MULENOK	526	738	58.36%
ANVAR SAMOST	1,547	2,073	57.26%
JUKU-KALLE RAID	243	321	56.91%
TÕNIS PALTS	99	127	56.19%
JEVGENI KRIŠTAFOVITŠ	266	327	55.14%
MART NUTT	179	214	54.45%
YOKO ALENDER	737	869	54.10%
AIVAR SÕERD	72	83	53.54%

Table 19: EP2014: Candidates with the ten highest i-vote proportions

4.23 General statistics

4.23.1 Age distribution

The youngest person who (unsuccessfully) attempted i-voting was seven years old, and the oldest i-voter was 103. The youngest vote verifier was 18 and the oldest was 93 (97 in KOV2013).

Voter turnout by age is shown in Figure 20. We see that the most active voters are people aged 30-40.

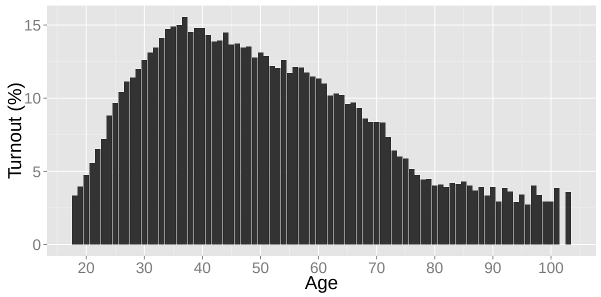


Figure 20: EP2014: Voter activity by age

The percentage of voters by age who verified their vote is shown in Figure 21. We see that voters older than 50 are less likely to verify their vote (in KOV2013 the verifier age distribution was more uniform).

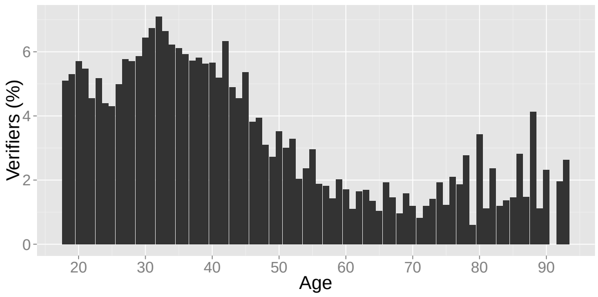


Figure 21: EP2014: Verifier activity by age

Similarly to KOV2013 we observed that older people are faster voters. This is illustrated in Figure 22.

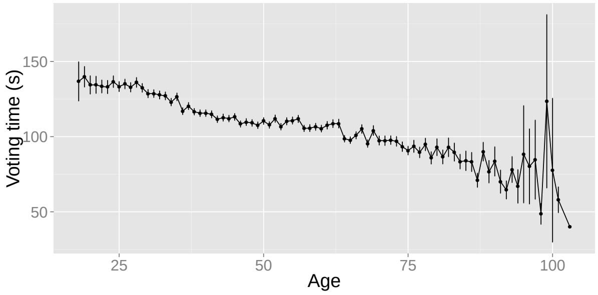


Figure 22: EP2014: Age vs voting time

4.23.2 Gender distribution

It has been observed for the last elections that more votes are cast by females. EP2014 was no exception. From all successful voters 51.53% (52.2% in KOV2013) were females.

However, if we look at the turnout, we see that 11.55% of eligible males i-voted, while from all eligible females 9.84% i-voted. Thus we see that in EP2014 males were 1.71% (1.16% in KOV2013) more active than females.

Figure 23 shows the percentage of female voter activity by age. We see that male and female activity is quite uniform.

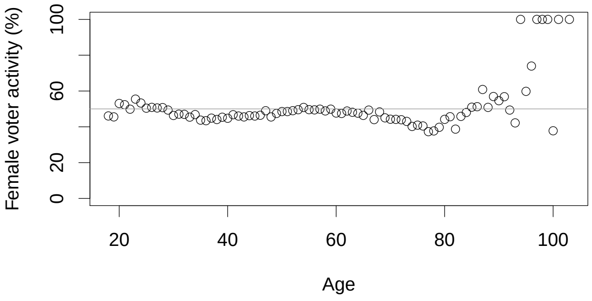


Figure 23: EP2014: Female voter activity by age

From all 4,250 verifiers only 1,120 (26.35%) (31.6% in KOV2013) were female. We see that 6.26% of male voters and 2.11% of female voters verified their vote. Thus, in EP2014, male voters were 2.96 (2.38 in KOV2013) times more active as verifiers than female voters. Figure 24 shows the percentage of female verifier activity by age.

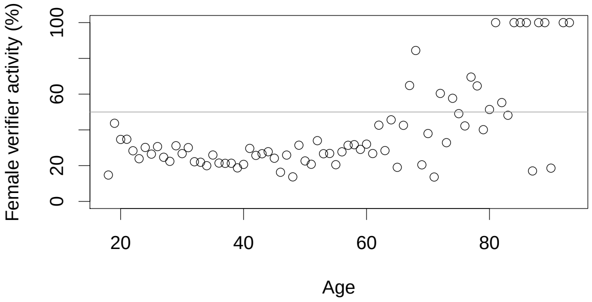


Figure 24: EP2014: Female verifier activity by age

4.23.3 OS distribution

From all the successfully cast votes (excluding votes annulled by revoting) the most popular OS was Windows at 93.4% (93.87% in KOV2013), then Mac at 5.46% (5.35%in KOV2013), and finally Linux at 1.14% (0.78% in KOV2013). OS distribution by age is shown in Figure 25. OS distribution by gender is shown in Figure 26.

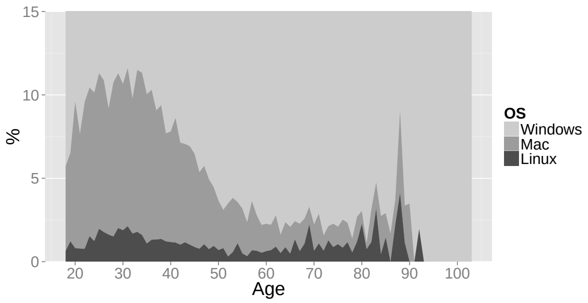


Figure 25: EP2014: OS distribution by age

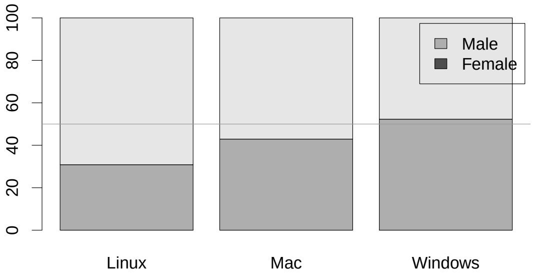


Figure 26: EP2014: OS distribution by gender

4.23.4 eID tool

From all the successfully cast votes (excluding votes anulled by revoting) the most popular eID tool was the ID card at 87.69% (90.27% in KOV2013), then Mobile-ID at 10.86% (8.49% in KOV2013), and finally Digi-ID at 1.45% (1.23% in KOV2013). eID distribution by age is shown in Figure 27. There we can see that Mobile-ID is especially popular among 30-year-olds. eID distribution by gender is shown in Figure 28.

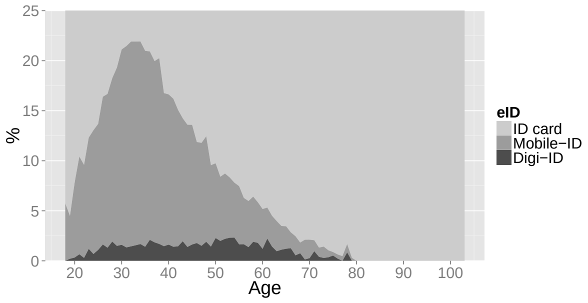


Figure 27: EP2014: eID distribution by age

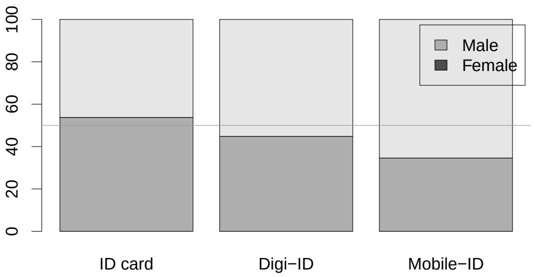


Figure 28: EP2014: eID distribution by gender

4.23.5 Verification

From all 4,250 verifiers 434 cast more than one i-vote, while 98 (22.58%) of them (19.85% in KOV2013) verified all their i-votes. From 336 verifiers who did not verify all their i-votes, 290 (86.31%) (85.2% in KOV2013) verified their last vote.

We see that Mobile-ID holders (voters who cast at least one vote using Mobile-ID) are 4.64 times (3.76 in KOV2013) more active verifiers than non-holders, since 13.68% of Mobile-ID holders verified their vote, while only 2.95% non-holders verified their vote.

The time between issuing the vote identifier and receiving the vote verification request is called "verification length". A voter can verify the same vote a total of three times but only within 60 minutes (30 minutes in KOV2013) after submitting the vote and before casting a new i-vote. Frequencies of verification lengths (taking into account only the first verification request made by the voter) are shown in Figure 29.

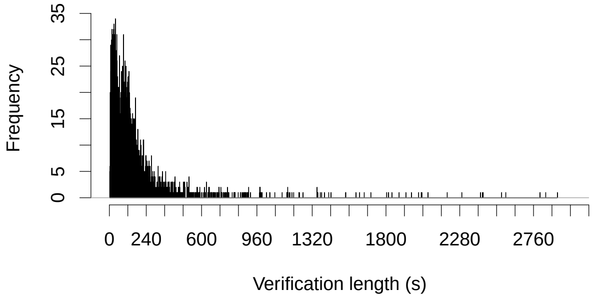


Figure 29: EP2014: Distribution of verification lengths

Table 20 shows how many times (at least) the voters verified their first vote and the corresponding success rate for consecutive verifications.

Times	1	2	3	4	5	6	7	8	9	10
Voters	4,250	401	110	76	55	38	29	20	13	12
Success	99.93%	92.77%	78.18%	–	–	–	–	–	–	–

Table 20: EP2014: Distribution of verification counts

We see that most voters do not perform more than one verification.

4.23.6 Verification OS distribution

Starting from EP2014 the verification application has been available also for Windows Phone and iPhone. The OS of the verification application is also logged on the server side.

Taking into account the verification application used for the voter's last verification, we see that the most popular verification OS was Android at 62.54% then iPhone at 28.12%, and finally Windows at 9.34%. Verification OS distribution by age is shown in Figure 30. We see that the choice of the mobile device OS does not depend on the verifier's age.

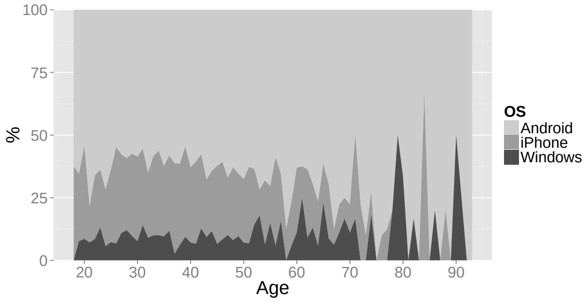


Figure 30: EP2014: Verification OS distribution by age

Verification OS distribution by gender is shown in Figure 31. We see that in both gender groups mobile device OS preference is almost equal.

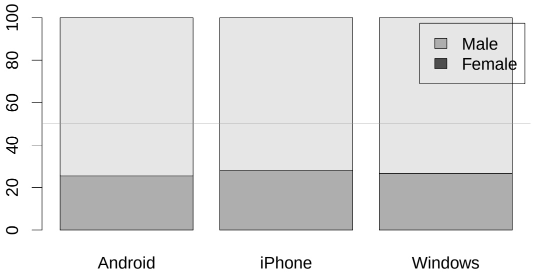


Figure 31: EP2014: Verification OS distribution by gender

The level of detail in an OS as reported by the verification application allows us to provide more detailed version statistics for Android (Table 21) and iPhone (Table 22) OS-based mobile devices.

Version	Voters
4.4.2	572
4.4	1
4.3.1	13
4.3	431
4.2.2	282
4.2.1	31
4.1.5	1
4.1.2	648
4.1.1	97
4.0.4	199
4.0.3	76
3.2.1	2
3.2	3
2.3.7	49
2.3.6	61
2.3.5	30
2.3.4	78
2.3.3	59
2.2.2	11
2.2.1	12
2.2	1

Table 21: EP2014: Android versions

Version	Voters
iOS 7.1.1	931
iOS 7.1	111
iOS 7.0.3, 7.0.4, 7.0.5, 7.0.6	76
iOS 6.1.2, 6.1.3, 6.1.4	51
iOS 7.0, 7.0.1, 7.0.2	9
iOS 5.1	8
iOS 6.0, 6.0.1	6
iOS 5.0.1	3

Table 22: EP2014: iPhone versions

4.24 Other irregularities

4.24.1 A public call to stop i-voting

Just three days before the start of i-voting in EP2014 an international team of researchers held a press conference calling Estonia to immediately withdraw the i-voting due to a major security risks being identified [14]. Since the findings of the team mainly emphasized the risks that have been accepted from the beginning of i-voting and the team did not contact the NEC before their public announcement, the actions of the team were perceived mainly as a reputation attack against the particular voting method used in Estonia. However, the report also included a number of constructive procedural remarks that have been taken into account for RK2015.

5 Results from RK2015

The i-voting in RK2015 (2015 Riigikogu elections [15]) took place from 19 February 2015 at 09:00 to 25 February 2015 at 18:00.

5,604,145 log entries from 19 February 2015 at 09:00:21 to 25 February 2015 at 18:37:11 were analysed. The starting point of the analysis was the moment when the voting period actually started. The ending point of the analysis was the moment when no more votes were accepted.

The breakdown of sessions, the number of unique voters connected to these sessions and the number of voters who did not manage to successfully i-vote (column "Voters (u)") is given in Table 23.

Session kind	Sessions	Voters	Voters (u)
All sessions	211,215	–	–
Voting	201,811	179,262	2,771
Successful	181,084	176,491	0
ID card	159,000	155,267	0
Mobile-ID	22,084	21,307	0
Unsuccessful	20,727	15,007	2,771
ID card	14,328	11,226	2,366
Mobile-ID	6,399	3,864	422
Verification	9,404	7,563	41
Successful	8,439	7,522	0
Unsuccessful	965	120	41

Table 23: RK2015: Session breakdown

5.1 Unexpected log entries

In total 67 voting sessions and 615 verification sessions raised an incident caused by unexpected log entries. Here we provide a grouped summary of them.

5.1.1 Inaccessible voter list

We observed an incident message stating that a vote submission request has been received from a voter who is ineligible. Since the eligibility of a voter is verified also on the candidate list request and there the voter was eligible, this was clearly an anomaly. The voter revoted successfully a few minutes later.

It was found that the anomaly was caused by a voter list update procedure performed at that time, which resulted in the voter list database being unreadable for a moment.

5.1.2 Vote submitted by a different person

Similarly as in EP2014 (see Section 4.1.5) we observed two ID card voting sessions where the credentials used to submit the vote were not of the same person who had obtained the candidate list.

In the first case the candidate list was obtained by a 37-year-old female, but the vote was submitted by a 63-year-old male. The female had already submitted a successful vote a few minutes earlier. The male submitted a successful vote a few minutes later.

In the second case the candidate list was obtained by a 59-year-old female, but the vote was submitted by a 84-year-old female. The 59-year-old female successfully revoted ten minutes later and then after a minute a successful vote was also cast by the 84-year-old female.

This can be explained by some voters forgetting to change the ID card before initiating a voting session for another voter.

5.1.3 Invalid ID card signature

We observed four ID card voting sessions initiated by four different voters using four different IP addresses and all using Windows IVCA, which failed with an incident message stating that the signature of the vote was invalid. All voters revoted successfully a few minutes later using the same ID card authentication certificate; however, in three of these cases the successful vote was signed using a different ID card digital signature certificate. The hash of the digital signature certificate used in the failed voting sessions could not be found in any other voting sessions.

The incidents were similar to these observed in KOV2013 (see Section 3.1.4). However, without the corresponding invalid votes, we were not able to investigate what caused these incidents.

5.1.4 Invalid signature of an ID card signing certificate

Similarly to KOV2013 (see Section 3.1.5), we observed an ID card voting session using Windows IVCA, which failed with an incident message stating that the certificate used to sign the vote had an invalid signature.

However, a few minutes later the voter successfully revoted using the same ID card authentication certificate, but a different ID card digital signature certificate. The hash of the digital signature certificate used in the failed voting session could not be found in any other voting session.

As in KOV2013, without the corresponding invalid vote, we were not able to investigate what caused this incident.

5.1.5 Invalid Mobile-ID authentication poll request

We observed 59 Mobile-ID voting sessions involving 26 unique voters that failed with an incident message stating that the Mobile-ID authentication poll request was malformed. These requests were traced back to the IVCA from EP2014 that used a different Mobile-ID authentication poll request format. Since the same server certificate hardcoded in the IVCA was used in both EP2014 and RK2015, the voters were able to use the IVCA from EP2014. However, since the election identifiers were different, even the voters who used an ID card or Digi-ID to cast the vote, received an error message after the candidate list was processed by the EP2014 IVCA. From these 26 voters 25 later tried again using the RK2015 IVCA.

5.1.6 Invalid verification request

We observed 615 verification requests querying 136 unique vote identifiers of votes given by 104 voters, which failed with an incident message stating that the verification request was malformed.

These requests were traced back to an old version of the vote verification application using a verification request not compatible with RK2015. Apparently these voters did not accept the automatic update offered by their mobile device. From the 104 voters affected 62 voters later managed to successfully verify their vote using the RK2015 VVA.

5.2 Incorrect session state change

We observed two ID card and 17 Mobile-ID voting sessions where more than one vote was submitted in a single voting session triggering an incident about illogical session state change.

Similarly to EP2014 (see Section 4.2), this behaviour was caused by a failed connection or timeout when waiting for a response to the vote submission request or Mobile-ID poll request and the possibility introduced in the IVCA since EP2014 to resubmit the vote.

5.3 Unsuccessful voting sessions

In the normality profile we defined that a voting session should end with a successfully cast vote. In practice out of 201,811 voting sessions 20,727 (10.27%) voting sessions (19.88% in KOV2013, 8.39% in EP2014) involving 15,007 voters did not result in a successfully cast vote.

The breakdown of error conditions, the number of unique voters affected in these voting sessions and the number of voters who did not manage to successfully i-vote (column “Voters (u)”) is given in Table 24. The Table 25 further details issues specific to Mobile-ID.

Reason for failure	Sessions	Voters	Voters (u)
Unsuccessful voting sessions	20,727	15,007	2,771
Explicit error	5,513	3,405	826
Common error	1,509	1,289	404
Maintenance	1	1	0
Under-aged voter	30	30	27
Ineligible voter	507	307	294
Voting ended	2	2	1
No new voters	87	77	54
Session expired	882	877	31
Certificate issue	641	298	271
ID card	572	298	271
Mobile-ID	69	–	–
Pre-2011 Mobile-ID user	366	249	89
Bad Mobile-ID number	974	–	–
DigiDocService failure	0	0	0
Mobile-ID failures	1,956	1,553	70
Incident	67	34	3
Other reason	15,214	12,072	2,009
Discontinued (Mobile-ID)	1,454	1,039	68
Authentication	1,008	731	51
Signing	446	415	20
Abnormal	0	0	0
Vote not submitted	13,760	11,103	1,947
ID card	12,283	9,779	1,744
Mobile-ID	1,477	1,353	206

Table 24: RK2015: Failed voting sessions

Some unsuccessful voting sessions (5,513 sessions, 3,405 voters) failed with an explicit error condition. From the 3,405 voters involved 826 voters did not manage to successfully i-vote.

Reason for failure	Sessions	Voters	Voters (u)
Mobile-ID failures	1,956	1,553	70
User cancelled	640	562	21
Authentication	355	306	15
Signing	285	274	6
Not in coverage	79	61	6
Authentication	73	56	6
Signing	6	6	0
SIM error	73	53	1
Authentication	41	28	1
Signing	32	31	0
SMS sending error	8	5	0
Authentication	8	5	0
Signing	0	0	0
Other	1,156	950	50
Authentication	539	413	32
Signing	617	580	20

Table 25: RK2015: Mobile-ID failures

In the largest portion of unsuccessful voting sessions (13,760 sessions, 11,103 voters) the candidate list was successfully downloaded, but the vote submission request did not follow. From these 11,103 voters 1,947 voters did not manage to cast their i-vote. From these 1,947 voters 43 voters had at least one voting session that failed. From the remaining 1,904 voters, 1,626 voters (2,000 in KOV2013, 700 in EP2014) had carried out a single voting session that did not continue after candidate list retrieval, 167 voters had two such sessions, and ten voters had more than six such sessions. We also observed two voters (nine in KOV2013, three in EP2014) who obtained the candidate list more than 15 times in a row and then cast their vote in the last voting session.

Some unsuccessful voting sessions were Mobile-ID sessions that were discontinued in the Mobile-ID authentication or signing phase. This could have been caused by a software error or a user closing the IVCA in the middle of the process.

There were no abnormal session interruptions in RK2015.

In 1,049 cases it was not possible to identify the voter associated with the unsuccessful voting session. These cases were exclusively Mobile-ID voting sessions and the vast majority of those (974) were due to the fact that the phone number was not associated with the Mobile-ID capable SIM card.

From the 179,262 persons who attempted to i-vote in RK2015, 176,491 (98.45%) (96.59% in KOV2013, 98.54% in EP2014) succeeded to cast at least one succesful vote.

5.4 Unsuccessful verifications and verification sessions

From all voters 7,604 (4.31%) (3.39% in KOV2013, 4.12% in EP2014) attempted to verify their i-vote.

In RK2015 the NEC received no complaints about unsuccessful vote verification.

However, we see that from 9,404 verification requests 965 (10.26%) (5.97% in KOV2013, 13.78% in EP2014) were unsuccessful.

The breakdown of reasons, the number of unique verifiers affected in these unsuccessful verification sessions and the number of verifiers who did not manage to successfully verify any vote (column "Verifiers (u)") is given in Table 26.

Reason for failure	Sessions	Verifiers	Verifiers (u)
Unsuccessful sessions	965	218	82
Newer vote cast	17	6	1
Verification count exceeded	154	63	6
Verification time exceeded	121	63	40
Invalid verification request	615	104	42
Vote ID not issued	58	–	–

Table 26: RK2015: Unsuccessful verification sessions

Most verification failures were caused by voters trying to verify the same vote more than three times or after the time allowed for vote verification had passed.

If we look at a voter's first verification attempt, we see that for 148 (33 in KOV2013, 26 in EP2014) voters their first verification attempt was not successful, resulting in an error message shown to the voter (43 – tried to verify after 30 minutes, 1 – after submitting a newer vote, 104 – used an outdated verification application (see Section 5.1.6)).

It is interesting to note that nine (ten in KOV2013, ten in EP2014) voters made their first verification request six hours after submitting the vote, five voters (six in KOV2013, six in EP2014) even a day after. Most likely these verifiers faced problems when installing the verification application.

We believe that these voters did not contact the NEC because they suspected that the verification failure was caused by their verification peculiarities. The case of four complaints received in EP2014 shows that if the voters had received an unjustified error message at least some of them would have contacted the NEC.

We observed a total of 58 vote verification requests for two unique vote identifiers (three in KOV2013, five in EP2014) that were not issued in the RK2015 elections and also not in KOV2013 and EP2014. The first vote identifier was queried by 31 IPs, in KOV2013 by 24 and in EP2014 by 15 IPs. The second vote identifier was queried by 1 IP, not queried in KOV2013, but in EP2014 it was queried by 87 IPs. These requests are again most likely made by curious people trying to verify the QR codes from the test elections.

5.5 Support requests handled by the NEC support centre

In RK2015 the NEC support centre registered 331 support requests. The breakdown by topics is shown in Table 27.

Topic	#
ID card certificates expired	2
Pre-2011 Mobile-ID user	2
Outdated ID-software	8
Website-related	12
General election questions	22
Mac OS X 10.7 and older not supported	47
Windows XP not supported	54
ID-software, card reader drivers	75
Other	109

Table 27: RK2015: Support requests handled

Mac OS X version 10.7 and older versions were not supported by the IVCA. In some cases it was possible to use a manual workaround by downloading the necessary driver and configuring the environment variable by hand.

The reason why so many Windows XP users complained was that in RK2015 Windows XP was not supported anymore; however, instead of providing a clear error message the IVCA crashed.

Other topics included IVCA Internet connectivity issues, questions on if it was possible to vote using a smart phone, tablet and a virtual mouse, or proxy usage, and situations where the person had a problem but did not provide enough information to diagnose it.

5.6 Voting sessions too slow

In normality profile we stated that a voting session should be completed in a few minutes. Figure 32 shows the histogram of actual voting session lengths observed in RK2015.

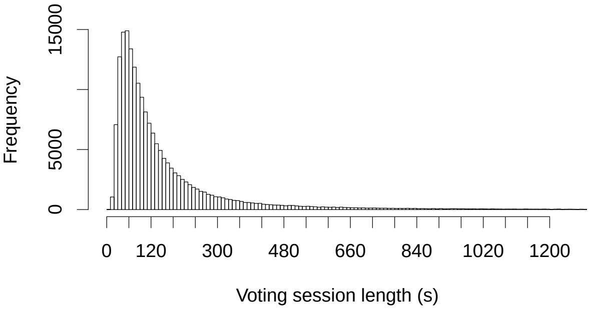


Figure 32: RK2015: Distribution of the voting session lengths

Minimal and maximal session lengths were 9 seconds and 479,400 (about 5.5 days), respectively. The mean length was 193 seconds and median length 96 seconds. 0.5%, 1%, 99% and 99.5% quantiles are given in Table 28.

Quantile	0.5%	1%	99%	99.5%
Value (s)	20	22	1,376	2,035

Table 28: RK2015: Quantiles of voting session lengths

The table allows us to estimate that the normal length for a voting session could be between 20 seconds and 23 minutes (20 minutes in KOV2013, 13 in EP2014). Note that for 91.04% voting sessions the session was shorter than six minutes. Note that there were 2336 sessions that were longer than 20 minutes, 350 sessions that were longer than an hour, 114 sessions that were longer than two hours, 29 sessions that were longer than six hours, nine sessions that were longer than a day.

In RK2015 session timeout enforcement was implemented on the server side. If the time between the candidate list request and the vote submission request was longer than 30 minutes, the submitted vote was rejected with an error message.

The slowest voting session, which took 5.5 days, was carried out by a male born in 1971, who used Windows OS and Mobile-ID to cast the vote. The candidate list was obtained from an IP in Estonia, but the vote was submitted from an IP in Austria. After receiving a timeout error message the voter did not try to revote.

The two fastest voting sessions (9, 10 seconds) were made by a NEC employee who was testing the voting system.

5.7 Vote signed with a different eID tool

There were no such sessions observed.

5.8 IP address or OS change in the middle of a voting session

We observed 177 voting sessions involving 177 unique voters (72 in KOV2013, 46 in EP2014) where the vote submission IP address was different from the candidate list retrieval IP address. The sessions were timewise evenly distributed over the i-voting period and the OS in these sessions did not change.

In 121 sessions the IP changed from one Estonian IP to another IP in Estonia. In 49 sessions the IP changed from one foreign IP to another IP in the same country and ISP. In seven voting sessions the IP changed from one country to another: from Finland to Estonia in 28.70 minutes, from Germany to the Netherlands in 2.03 minutes, from Estonia to Finland in 2.13 hours, from Estonia to Austria in 5.55 days, from Estonia to the US in 1.18 minutes, from Estonia to Germany in 7.83 minutes, from Korea to China in 12.78 minutes.

5.9 IP address shared by several voters

In the RK2015 elections 176,491 voters used 83,431 unique IP addresses to cast their successful votes, which means that in RK2015 on average 2.11 persons (1.95 in KOV2013, 1.97 in EP2014) shared one IP address.

There were 28 IP addresses (28 in KOV2013, 22 in EP2014) that were each shared by more than 100 voters with the top IP shared by 1,415 voters (1,127 in KOV2013, 970 in EP2014). We reviewed the top shared voting IPs and did not notice any strange patterns – voting was evenly distributed over the voting period, different OS versions were used and several voting sessions overlapped.

We observed a large number of IP addresses shared by two and more voters where the voting sessions were not evenly distributed over the voting period, with the voters casting their votes shortly after each other. Table 29 shows the number of voter groups observed, where voters voting with five-minute intervals and using the same OS are considered to be one group. The table contains data only about these IP addresses that do no have overlapping voting sessions and these whose first and last voting activity falls within a 24-hour window.

Voters	Groups
2	10,795
3	1,045
4	150
5	15
6	1
7	1

Table 29: RK2015: Voter groups

The voters from the group of seven voters all voted using an ID card from an IP address in Colombia and took 20 minutes to vote in total.

5.10 Non-unique vote encryption

All of the votes received had a unique ciphertext.

5.11 Large percentage of revoters

From 176,491 voters 4,034 (2.29%) voters (1.93% in KOV2013, 1.69% in EP2014) cast more than one vote. From these revoters 3,723 voted two times, 254 voted three times, and 57 voted four times or more.

The distribution of time between the revoters' first and second vote is shown in Figure 33.

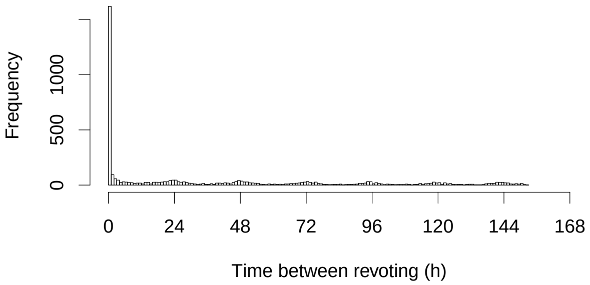


Figure 33: RK2015: Distribution of time between revotes

We can see that 29% (30% in KOV2013, 28% in EP2014) of revoters revote in the first ten minutes, and 40% (41% in KOV2013, 38% in EP2014) of revoters revote in the first hour after casting their vote.

Figure 34 shows the distribution of votes and revotes over the voting period. We see that revotes are evenly distributed over the voting period.

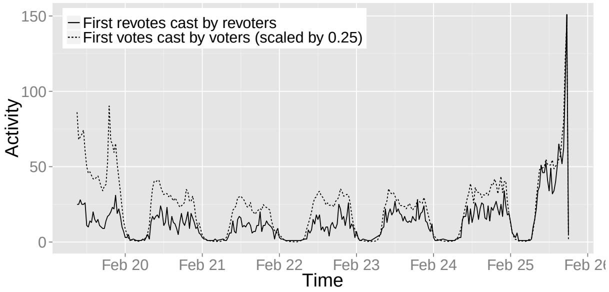


Figure 34: RK2015: Distribution of votes and revotes

We can estimate that in the worst case in the RK2015 elections 4,034 votes (2,586 in KOV2013, 1,743 in EP2014) could have been replaced by a revoting malware described in Section 2.7.11.

However, since in the previous elections revoter proportion was similar (see Section 2.7.11, 3.11, 4.11) and some amount of revoters is normal, it is unlikely that most of the revotes would have been caused by an attack.

5.12 Voters revoting many times

The top 10 revoters cast 60, 37, 29, 19, 12, 11, 10, 10, 8, 8 number of votes.

5.13 Revoting using a different eID tool

In total 92 revoters used more than one eID tool to cast their vote. In 48 of these cases the voter also used a different IP address to revote.

5.14 Revoting using the same eID tool but different certificates

We observed 27 voters who had at least two voting sessions using the same type of eID tool, but with eID tool containing different certificates. In the case of 19 voters we see that the first voting sessions failed with a revoked certificate error and finally the vote was successfully cast using a different certificate. From the remaining eight cases in three cases Mobile-ID voters revoted on another day with different certificates from a different IP address. In one case the voter revoted 40 hours later with different ID card certificates from the same IP address. In four cases only the digital signature certificate changed and the voting sessions raised an incident (three of the cases are described in Section 5.1.3 and one case described in Section 5.1.4). Thus we can conclude that in all cases (except the four unclear cases where only the digital signature certificate changed) the certificate changes were caused by certificate renewal and not an NCA compromise.

5.15 Revoting from different IP addresses

In total 948 revoters (23.50%) (20.84% in KOV2013, 20.54% in EP2014) revoted from a different IP address. From these 948 revoters 47 revoted from an IP in a different country. In 44 cases the same eID tool was used to revote. In all the cases except one the time difference between revotes was long enough for the voter to physically change her country of location.

5.16 Parallel voting sessions

We observed 99 voters (60 in KOV2013, 28 in EP2014) who had parallel voting sessions. In all cases the voting session was carried out using the same eID tool from the same IP address and with the same OS.

5.17 Vote verified from different IP addresses

There were 49 votes (19 in KOV2013, 23 in EP2014) which were verified from more than one IP address. From these 49 votes, 44 votes were verified from two different IP addresses and in three cases had a different verification application OS, two votes were verified from three different IP addresses using the same verification application OS. However, three votes were verified from more than three IP addresses (4, 7 and 8 different IPs) using different verification application OSs and the verifications were performed over a period of several days. We believe that in these three cases the verification QR code was published on the Internet.

We found that the vote identifiers verified from four and eight different IP addresses were published on the Internet by two politicians [16] and [17] respectively. Luckily, since the verification requests were made 30 minutes after the vote was cast, none of the verification requests succeeded.

5.18 Voter's votes verified from different IP addresses

There were 104 voters (67 in KOV2013, 63 in EP2014) whose votes were verified from more than one IP. A summary of these voters aggregated by the number of different verification IPs can be seen in Table 30.

Verification IPs	2	3	4	5	7	8
Voters	87	11	3	1	1	1
Max (votes)	37	60	8	11	1	1

Table 30: RK2015: Summary of voters whose votes were verified from different IP addresses

From these revoters, three displayed verification activity that could indicate an attack described in Section 2.7.18. One who revoted 37 times and whose votes were verified from two different IPs, one who revoted 60 times and whose votes were verified from three different IPs and one who revoted 11 times and whose votes were verified from five different IPs. However, these three are unlikely to be attackers, since the OS used to verify the votes did not change. Since the attacker does not know which mobile device OS the verifier has, the verification OSs seen in an attack would be distributed according to mobile device OS popularity.

We can also exclude other revoters due to the negligible number of votes cast by these voters. An attacker who would want to successfully attack three verifiers without being detected would need at least ten times more revotes since less than 5% of voters verify their vote.

5.19 IP address shared by several verifiers

There were 1,425 IP addresses (746 in KOV2013, 721 in EP2014) which were each shared by several verifiers. The IP addresses were each shared by 11 (ten in KOV2013, 13 in EP2014) and fewer verifiers.

In the RK2015 elections 7,604 verifiers used 5,438 unique IP addresses to verify their votes, which means that in RK2015 on average 1.4 (1.35 in KOV2013, 1.31 in EP2014) persons shared one verification IP address.

We see that in the RK2015 elections 60.17% (53.28% in KOV2013, 56.82% in EP2014) of verifiers verified their vote from the same IP address that was used to cast the vote.

5.20 First voting session seen as revoting

No cases have been registered by the NEC.

5.21 Non-i-voter denied paper vote

No cases have been registered by the NEC.

5.22 I-voting results deviating from paper voting results

In the RK2015 elections 30.5% [2] of votes were i-votes. Thus, on average a candidate received 30.5% i-votes.

Table 31 shows candidates who have received the highest proportion of i-votes and who have received at least 30 i-votes. The proportion is close to other candidates and is not extreme enough to raise suspicion.

Candidate	p-votes	i-votes	i-votes (%)
MATI SAREVET	22	38	63.33%
LINDA EICHLER	49	84	63.15%
REIGO KIMMEL	35	60	63.15%
TARMO KALDMA	37	60	61.85%
EGGE KULBOK-LATTIK	34	55	61.79%
SIIM TUISK	103	165	61.56%
TOOMAS VIKS	37	58	61.05%
SIRJE KEEVALLIK	66	100	60.24%
MONIKA HAUKANÕMM	181	269	59.77%
GEORG AHER	70	104	59.77%

Table 31: RK2015: Candidates with the ten highest i-vote proportions

5.23 General statistics

5.23.1 Age distribution

The youngest person who (unsuccessfully) attempted i-voting was nine years old, and the oldest i-voter was 104. The youngest vote verifier was 18 and the oldest was 95 (97 in KOV2013, 93 in EP2014).

Voter turnout by age is shown in Figure 35. We see that the most active voters are people aged 30-40.

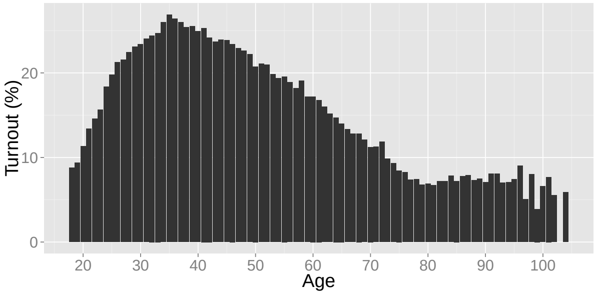


Figure 35: RK2015: Voter activity by age

The percentage of voters by age who verified their vote is shown in Figure 36. We see that voters older than 50 are less likely to verify their vote (to a greater extent compared to KOV2013, but to a lesser extent compared to EP2014).

Similarly to KOV2013 and EP2014 we observed that older people are faster voters. This is illustrated in Figure 37.

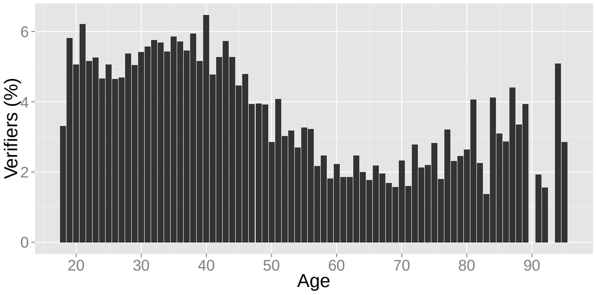


Figure 36: RK2015: Verifier activity by age

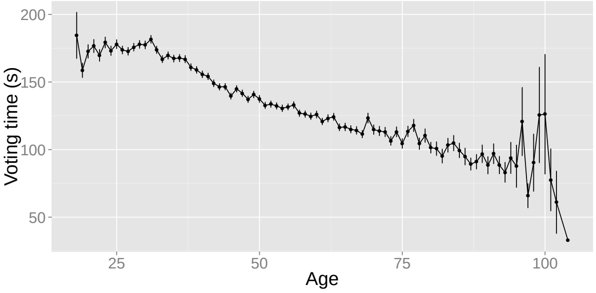


Figure 37: RK2015: Age vs voting time

5.23.2 Gender distribution

It has been observed for the last elections that more votes are cast by females. RK2015 was no exception. From all successful voters 52.65% (52.2% in KOV2013, 51.53% in EP2014) were females.

However, if we look at the turnout, we see that 19.54% of eligible males i-voted, while from all eligible females 17.4% i-voted. Thus we see that in RK2015 males were 2.14% (1.16% in KOV2013, 1.71% in EP2014) more active than females.

Figure 38 shows the percentage of female voter activity by age. We see that male and female activity is quite uniform.

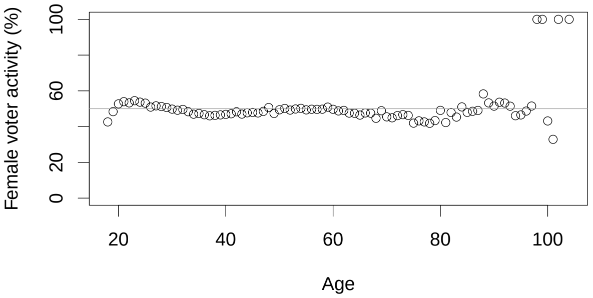


Figure 38: RK2015: Female voter activity by age

From all 7,604 verifiers only 2,454 (32.27%) (31.6% in KOV2013, 26.35% in EP2014) were female. We see that 6.16% of male voters and 2.64% of female voters verified their vote. Thus, in RK2015, male voters were 2.33 (2.38 in KOV2013, 2.96 in EP2014) times more active as verifiers than female voters.

Figure 39 shows the percentage of female verifier activity by age.

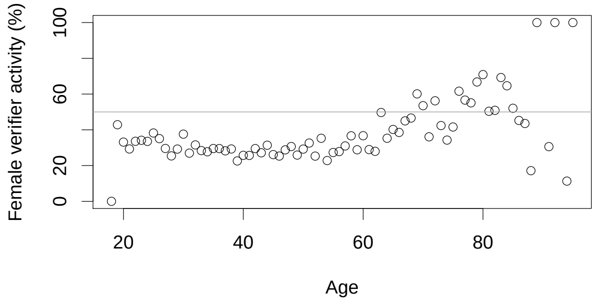


Figure 39: RK2015: Female verifier activity by age

5.23.3 OS distribution

From all the successfully cast votes (excluding votes annulled by revoting) the most popular OS was Windows at 92.71% (93.87% in KOV2013, 93.4% in EP2014), then Mac at 6.18% (5.35% in KOV2013, 5.46% in EP2014), and finally Linux at 1.12% (0.78% in KOV2013, 1.14% in EP2014). OS distribution by age is shown in Figure 40. OS distribution by gender is shown in Figure 41.

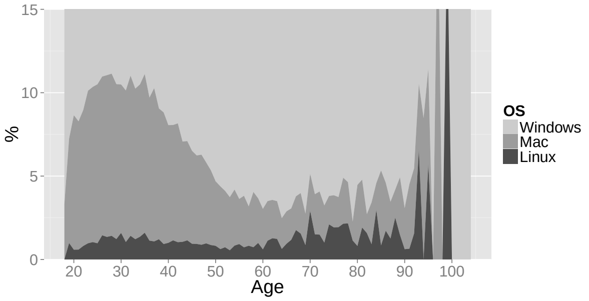


Figure 40: RK2015: OS distribution by age

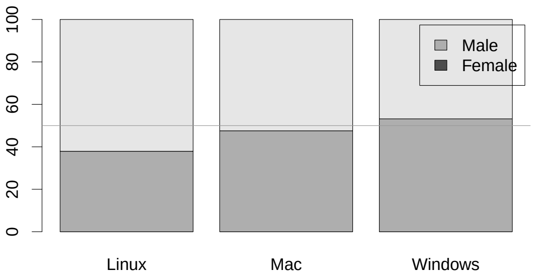


Figure 41: RK2015: OS distribution by genders

5.23.4 eID tool

From all the successfully cast votes (excluding votes anulled by revoting) the most popular eID tool was the ID card at 86.55% (90.27% in KOV2013, 87.69% in EP2014), then Mobile-ID at 12.05% (8.49% in KOV2013, 10.86% in EP2014), and finally Digi-ID at 1.4% (1.23% in KOV2013, 1.45% in EP2014). eID distribution by age is shown in Figure 42.

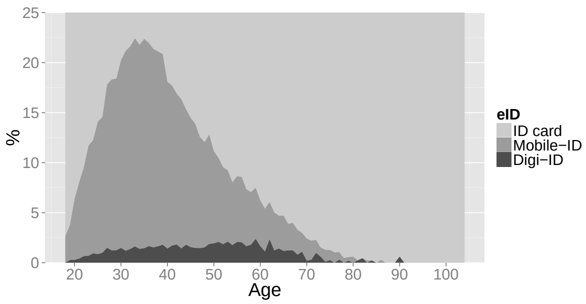


Figure 42: RK2015: eID distribution by age

We can see that Mobile-ID is especially popular among 30-year-olds. eID distribution by gender is shown in Figure 43.

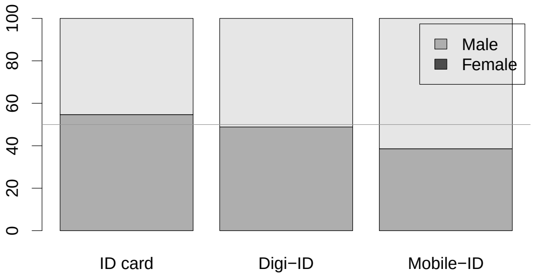


Figure 43: RK2015: eID distribution by gender

5.23.5 Verification

From 7,604 verifiers 798 cast more than one i-vote, while 143 (17.92%) of them (19.85% in KOV2013, 22.58% in EP2014) verified all their i-votes. From 655 verifiers who did not verify all their i-votes, 578 (88.24%) (85.2% in KOV2013, 86.31% in EP2014) verified their last vote.

We see that Mobile-ID holders (voters who cast at least one vote using Mobile-ID) are 3.84 times (3.76 in KOV2013, 4.64 in EP2014) more active verifiers than non-holders, since 12.32% of Mobile-ID holders verified their vote, while only 3.21% non-holders verified their vote.

The time between issuing the vote identifier and receiving the vote verification request is called "verification length". A voter can verify the same vote a maximum of three times but only within 30 minutes (30 minutes in KOV2013, 60 in EP2014) after the vote has been submitted and before the voter has cast a new i-vote.

Frequencies of verification lengths (taking into account only the first verification request made by the voter) are shown in Figure 44.

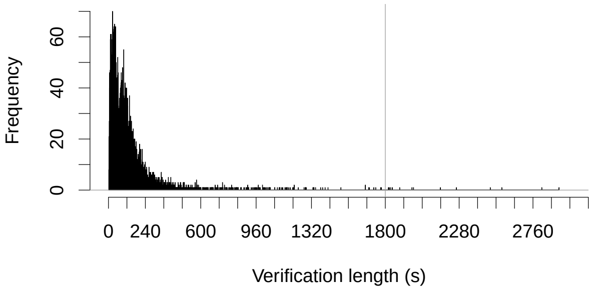


Figure 44: RK2015: Distribution of verification lengths

Table 32 shows how many times (at least) the voters verified their first vote and the corresponding success rate for consecutive verifications. We see that most voters do not perform more than one verification.

Times	1	2	3	4	5	6	7	8	9	10
Voters	7,604	737	202	121	79	55	40	32	26	21
Success	98.05%	82.36%	48.02%	–	–	–	–	–	–	–

Table 32: RK2015: Distribution of verification counts

5.23.6 Verification OS distribution

Taking into account the verification application used for the voter's last verification, we see that the most popular verification OS was Android at 65.08% then iPhone at 25.46%, and finally Windows at 9.46%. Verification OS distribution by age is shown in Figure 45. We see that the choice of the mobile device OS does not depend on the verifier's age.

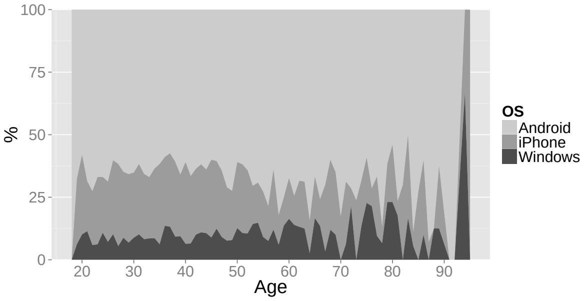


Figure 45: RK2015: Verification OS distribution by age

Verification OS distribution by gender is shown in Figure 46. We see that in both gender groups mobile device OS preference is almost equal.

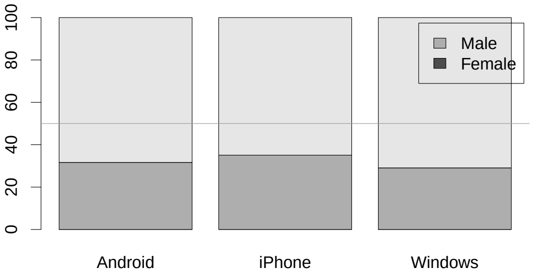


Figure 46: RK2015: Verification OS distribution by gender

The level of detail in an OS as reported by the verification application allows us to provide more detailed version statistics for Android (Table 33) and iPhone (Table 34) OS-based mobile devices.

Version	Voters
5.0.2	68
5.0.1	393
5.0	174
4.4.4	1,147
4.4.3	140
4.4.2	1,104
4.4	1
4.3.1	3
4.3	465
4.2.9	1
4.2.2	293
4.2.1	27
4.1.2	593
4.1.1	82
4.0.4	151
4.0.3	62
3.2	2
3.1	1
2.3.7	36
2.3.6	65
2.3.5	14
2.3.4	68
2.3.3	41
2.2.2	7
2.2.1	11

Table 33: RK2015: Android versions

Version	Voters
iOS 8.1.1, 8.1.2, 8.1.3	1,524
iOS 7.1.2	219
iOS 8.1.0	49
iOS 8.0, 8.0.1, 8.0.2	31
iOS 7.1.1	31
iOS 7.0.3, 7.0.4, 7.0.5, 7.0.6	29
iOS 6.1.2, 6.1.3, 6.1.4	27
iOS 7.1	13
iOS 5.1	4
iOS 6.0, 6.0.1	3
iOS 7.0, 7.0.1, 7.0.2	3
iOS 8.2	2
iOS 5.0.1	1

Table 34: RK2015: iPhone versions

5.24 Other irregularities

5.24.1 Invalid vote cast

Similarly to RK2011 [3, Section 3.1] and KOV2013 (Section 3.24.1) in the vote tallying process it was found that the encryption of one vote was invalid.

An activist from the Estonian Pirate Party took credit for casting the spoiled ballot [18]. The technique employed involved using a GNU debugger to locate the breakpoint in Linux IVCA where the candidate number is stored and replace it with an invalid candidate number.

5.24.2 Vote-buying accusations

Shortly after the RK2015 elections a newspaper published an article [19] about the manager of a pensioner day care centre in Võru who was accused of assisting pensioners to i-vote in favour of the candidate Inara Luigas. The police investigated the complaint but did not find any violations.

From the election results we see that Inara Luigas received 935 paper votes and 257 i-votes with the i-voting ratio being only 21.56%.

5.24.3 Older-than-average i-voters

Shortly after the elections the channel Tallinna TV, which is owned by the Tallinn City Government, ran a story claiming that the age of the RK2015 i-voters was suspicious [20]. The suspicious part was that in RK2015 90-year-old people were more active i-voters than 18-year-olds.

In the data we see that indeed in RK2015 there were 151 i-voters who were 18 years old and 162 i-voters who were 90 years old. While it would not be too surprising to see that older people are more active i-voters, if we look at the voters' age relative to the age of eligible voters (see Figure 6, 20 and 35) we see that in all elections, 18-year-olds were actually more active i-voters than 90-year-olds.

6 Summary of Findings

6.1 Attacks

We did not observe any event which could qualify as an attack against the i-voting system.

Furthermore, taking into account all observations, especially those described in Sections 2.7.11, 2.7.4 and 2.7.18, we can conclude that in KOV2013, EP2014 and RK2015 no large-scale attack was executed against the i-voters.

As an interesting proof-of-concept attack against the IVCA we can note a case in KOV2013 and RK2015 where, presumably by manipulating the IVCA, someone cast an invalid vote which did not contain a valid candidate number (Section 3.24.1, 5.24.1).

6.2 System malfunctions

6.2.1 System unavailability

In the KOV2013 elections 94 voting sessions failed because of either VSS or NCA DigiDocService downtime (Section 3.1.1, 3.3). In EP2014 no such voting sessions were observed. In RK2015 two voting sessions failed because of VSS maintenance (Section 5.3, 5.1.1).

6.2.2 IVCA

In KOV2013 several bugs affecting the IVCA were found. Two bugs in the IVCA which have been fixed (Section 3.1.2, 3.1.7), two bugs in external libraries (Section 3.3, 3.1.3) – one which was fixed in the library version bundled with the IVCA and one which affected voters also in EP2014 (Section 4.1.2), and two bugs whose cause could not be found (Section 3.1.4, 3.1.5).

In RK2015 were observed two possible bugs in the IVCA (similar to the ones observed in KOV2013) whose cause is still unknown (Section 5.1.3, 5.1.4).

6.2.3 VVA

In EP2014 two bugs in the iOS-based vote verification application were found. One bug that resulted in the acceptance of QR codes that were not created by the IVCA (Section 4.1.1) and one bug that caused a verification error in 7.53% of the cases (Section 4.4).

6.2.4 Server-side

In KOV2013 two bugs were discovered in the server-side code causing logging deficiencies (Section 3.3, 3.4).

In EP2014 one bug in the server-side code was found, which affected voters with recently renewed certificates (Section 4.1.3) and one, which caused a logging deficiency (Section 4.1.1).

In RK2015 a bug in the server-side code caused an incorrect error message returned to the voter at the time when the voter list was updated on the VSS (Section 5.1.1).

6.3 Voter behaviour

In EP2014 a bug was found in the iOS-based verification application that appeared as failed verification to the voter. From the estimated number of 42 voters who received the verification error, only four contacted the NEC (Section 4.4). This indicates that only 10% of voters are willing to inform the NEC about any verification irregularities they observe.

In KOV2013 and EP2014 voters' verification QR codes were verified only by a few devices, which may suggest that voters are cautious and do not share their verification QR codes (Section 3.17, 4.17). However, in RK2015 we observed three QR codes verified by many devices. Two of these QR codes were published on the Internet (Section 5.17). It is not known whether the voters were aware of the privacy risk associated with making the verification QR code available to other persons.

For unknown reasons 60 people in KOV2013, 28 people in EP2014 and 99 people in RK2015 ran parallel IVCA instances when casting their vote (Section 3.16, 4.16, 5.16).

Less than 3% of voters revote and 40% of the revoters revote within the first hour after casting their first vote (Section 3.11, 4.11, 5.11).

Around 20% of revoters revote from a different IP address (Section 3.15, 4.15, 5.15).

Apparently some voters are routing their Internet traffic through VPNs in foreign countries (Section 3.8, 4.8, 5.8).

Most of the voters cast their i-vote within two minutes (Section 3.6, 4.6, 5.6).

In all elections we observed a significant number of voters who did not submit a vote after obtaining the candidate list. For unknown reasons, some voters made a large number of candidate list requests before finally casting their vote (Section 3.3, 4.3, 5.3).

In EP2014 five voters and in RK2015 two voters, for unknown reasons, after obtaining the candidate list swapped their ID card with another person's ID card which was then used to sign the vote (Section 4.1.5, 5.1.2). In EP2014 three of these voters did not cast their own i-vote.

We see that in many cases voters sharing the same IP address vote shortly after each other, possibly sharing the same voting device (Section 3.9, 4.9, 5.9).

6.4 Other

In KOV2013 a voter with a defective Mobile-ID SIM card producing an invalid signature was found (Section 3.1.6).

In KOV2013 we observed eight and in RK2015 13 ineligible voters who in the middle of the i-voting period became eligible and i-voted (Section 3.3, 5.3).

In EP2014 we observed one and in RK2013 three voters who at the beginning of the i-voting period were younger than 18, but later reached 18 years of age and were able to i-vote (Section 4.3, 5.3).

We see that males are slightly more active voters and significantly more active verifiers (Section 3.23.2, 4.23.2, 5.23.2).

The most active voters are people aged 35 (Section 3.23.1, 4.23.1, 5.23.1).

Mobile-ID holders are more than three times more active verifiers than non-holders (Section 3.23.5, 4.23.5, 5.23.5).

Older voters perform the voting process faster (Section 3.23.1, 4.23.1, 5.23.1).

We see a trend indicating that Windows OS is slowly losing its popularity in favour of Mac and Linux (Section 3.23.3, 4.23.3, 5.23.3).

We see that Mac and (even more) Linux is a less popular OS for female voters (Section 3.23.3, 4.23.3, 5.23.3).

We see a trend indicating that Digi-ID and Mobile-ID are slowly becoming more popular tools for i-voting (Section 3.23.4, 4.23.4, 5.23.4).

We see that Digi-ID and (even more) Mobile-ID are less popular eID tools among female voters (Section 3.23.4, 4.23.4, 5.23.4).

We see that the mobile device OS used for vote verification is quite uniform between genders and verifiers of different ages (Section 4.23.6, 5.23.6).

On average two people share one IP address to cast a vote (Section 3.9, 4.9, 5.9).

On average 1.35 people share one IP address to verify a vote (Section 3.19, 4.19, 5.19).

More than 50% of voters use the voting IP address to verify their vote and we see a trend that this percentage is increasing (Section 3.19, 4.19, 5.19).

In KOV2013 the highest i-vote proportion for a candidate was 75.00%, in EP2014 61.75% and in RK2015 63.33% (Section 3.22, 4.22, 5.22).

7 Limitations and Discussion

The main limitation for our analysis is the limited ability to find the causes for some anomalies in the data.

In some of these cases the causes might be found if the voter could be contacted for an explanation (e.g., see Section 3.3, 4.3, 5.3 and 4.1.5, 5.1.2). However, there is no simple way to contact the voter[5] and there is no legal basis for it, unless there is convincing evidence that there could have been illegal activity. The only case when a voter was contacted was the case of the voter who cast more than 500 votes (see Section 2.7.12), and even then the inquiry did not provide a plausible explanation for the anomalous behaviour observed.

Some incidents could not be investigated because of technical reasons, such as the unavailability of the vote involved in an incident (e.g., see Section 3.1.4, 3.1.5, 5.1.3, 5.1.4). The logging and availability of such data for investigation is deliberately limited by the NEC due to vote secrecy concerns.

[5] Although if the voter used Mobile-ID to cast the vote, the phone number registered to the voter would be available to the NEC.

The investigation of some incidents which depend on external data related to a specific individual (e.g., see Section 4.1.4), if done after elections, can be complicated or even impossible, since after the end of the i-voting period the logs are pseudonymised before they are made available for more detailed analysis.

Obviously, the approach used in this work can be used to detect only the attacks executed by external attackers who attack the voters' voting devices or eID tools, since none of the anomaly patterns applied[6] can be used to detect large-scale vote manipulation attacks carefully executed by i-voting servers. Therefore, server-side attacks must be detected using different means.

As the i-voting server-side source code was published on GitHub [21], our log monitoring solution is unlikely to observe incidents caused by reconnaissance exploitation attempts against i-voting servers, since now the attacker does not have to develop her attacks on a live election system. The exploit can be developed using a cloned i-voting system fully operated by the attacker.

8 Conclusions

In this work we developed a systematic data analysis method that can be used to assess the state of an ongoing i-voting and to perform post-election analysis.

The log monitoring solution developed has been a useful tool for detecting software bugs and logging deficiencies, which might not have been otherwise detected. We note that in order to take full advantage of log analysis, the logging requirements have to be evaluated in the design process of an i-voting solution.

Although the three elections analyzed in this study were different types of elections, we can see that most of the measured values are similar. Furthermore, taking into account all the observations, we can conclude that in KOV2013, EP2014 and RK2015, no large-scale attack against i-voters was carried out.

The unexplained voter behaviour observed in the study gives an interesting starting point for further user studies.

[6] Perhaps except for the anomaly pattern described in Section 2.7.22.

Acknowledgements

We would like to thank Arne Koitmäe, Tanel Kuusk, Uve Lokk, Leino Mandre, Tarvi Martens, Tiit Pikma, Tavo Toomemägi and Priit Vinkel for the information provided for this report.

The research was supported by the Estonian Research Council under the Institutional Research Grant IUT27-1, the Estonian Doctoral School in Information and Communication Technology (IKTDK) and the European Regional Development Fund through the Centre of Excellence in Computer Science (EXCS) and grant project number 3.2.1201.13-0018 "Verifiable Internet Voting – Event Analysis and Social Impact".

References

[1] Sven Heiberg, Arnis Parsovs, and Jan Willemson. Log Analysis of Estonian Internet Voting 2013–2014. In Rolf Haenni, Reto E. Koenig, and Douglas Wikström, editors, *E-Voting and Identity*, volume 9269 of *Lecture Notes in Computer Science*, pages 19–34. Springer International Publishing, 2015.

[2] Estonian National Electoral Committee. Statistics about Internet Voting in Estonia, 2015. `http://vvk.ee/voting-methods-in-estonia/engindex/statistics`.

[3] Sven Heiberg, Peeter Laud, and Jan Willemson. The Application of I-Voting for Estonian Parliamentary Elections of 2011. In Aggelos Kiayias and Helger Lipmaa, editors, *VOTE-ID*, volume 7187 of *Lecture Notes in Computer Science*, pages 208–223. Springer, 2011.

[4] Sven Heiberg and Jan Willemson. Verifiable Internet Voting in Estonia. In Robert Krimmer and Melanie Volkamer, editors, *6th International Conference on Electronic Voting 2014, (EVOTE 2014), October 28-31, 2014, Bregenz, Austria*, pages 23–29. TUT Press, 2014.

[5] MaxMind. GeoLite Free Downloadable Databases, 2015. http://dev.maxmind.com/geoip/legacy/geolite/.

[6] Peeter Laud and Meelis Roos. Formal Analysis of the Estonian Mobile-ID Protocol. In Audun Jøsang, Torleiv Maseng, and Svein J. Knapskog, editors, *NordSec*, volume 5838 of *Lecture Notes in Computer Science*, pages 271–286. Springer, 2009.

[7] Christian Bull and Henrik Nore. Problems encountered. *Seminar on Internet voting*, September 2013. https://www.regjeringen.no/contentassets/c41c2959b8d946bf8007b546552ff9dc/5_problems_encountered.pdf.

[8] Véronique Cortier and Ben Smyth. Attacking and fixing Helios: An analysis of ballot secrecy. *Journal of Computer Security*, 21(1):89–148, 2013.

[9] Sven Heiberg and Jan Willemson. Modeling Threats of a Voting Method. In Dimitrios Zissis and Dimitrios Lekkas, editors, *Design, Development, and Use of Secure Electronic Voting Systems*, pages 128–148. IGI Global, 2014.

[10] Estonian National Electoral Committee. Municipal Elections 2013 Results, 2013. http://kov2013.vvk.ee/.

[11] OpenSC project. Regression in e35febe: compute cert length, December 2012. https://github.com/OpenSC/OpenSC/pull/114.

[12] Timeout for Expect: 100-continue as an option, October 2013. Curl-library mailing list archives, http://curl.haxx.se/mail/lib-2013-10/0142.html.

[13] Estonian National Electoral Committee. European Parliament Elections 2014 Results, 2014. http://ep2014.vvk.ee/detailed-en.html.

[14] International team of independent experts identifies major risks in the security of Estonia's Internet voting system, May 2014. https://estoniaevoting.org/press-release/.

[15] Estonian National Electoral Committee. Riigikogu Elections 2015 Results, 2015. http://rk2015.vvk.ee/.

[16] Eesti Reformierakond. Prime Minister of Estonia explains how fast, simple and safe is e-voting, February 2015. https://www.youtube.com/watch?v=yZ4s95lFkk4#t=107.

[17] Lauri Bambus. Twitter post: It took less than 1 minute to e-vote at Estonian Parliamentary 2015 election, February 2015. https://twitter.com/LauriBambus/status/568355079318835200/photo/1.

[18] Märt Põder. How I hacked a bit e-elections (in Estonian), March 2015. http://boamaod.github.io/blog/2015/03/02/minu-evalimised/.

[19] Postimees. Police investigated electoral fraud in Võru Senior Center (in Estonian), March 2015. http://www.postimees.ee/3111479/politsei-uuris-valimispettust-voru-pensionaride-keskuses.

[20] Tallinna TV. E-voting results rises a question about older than average voters (in Estonian), March 4, 2015.

[21] Estonian National Electoral Committee. Source code of the server side components of Estonian internet-voting system, July 2013. https://github.com/vvk-ehk/evalimine.

Appendix A: UML diagrams

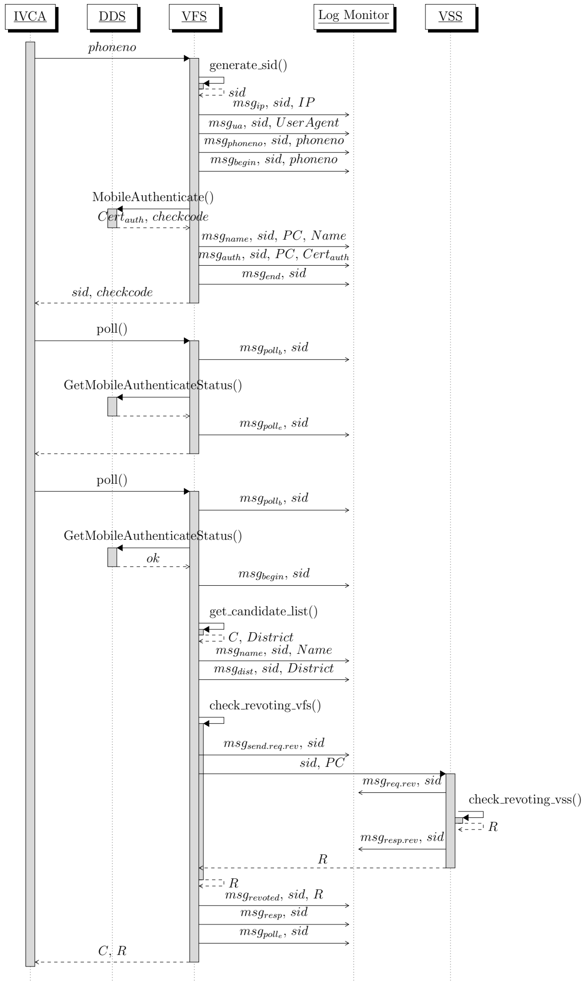


Figure 47: Logs generated on candidate list retrieval (Mobile-ID)

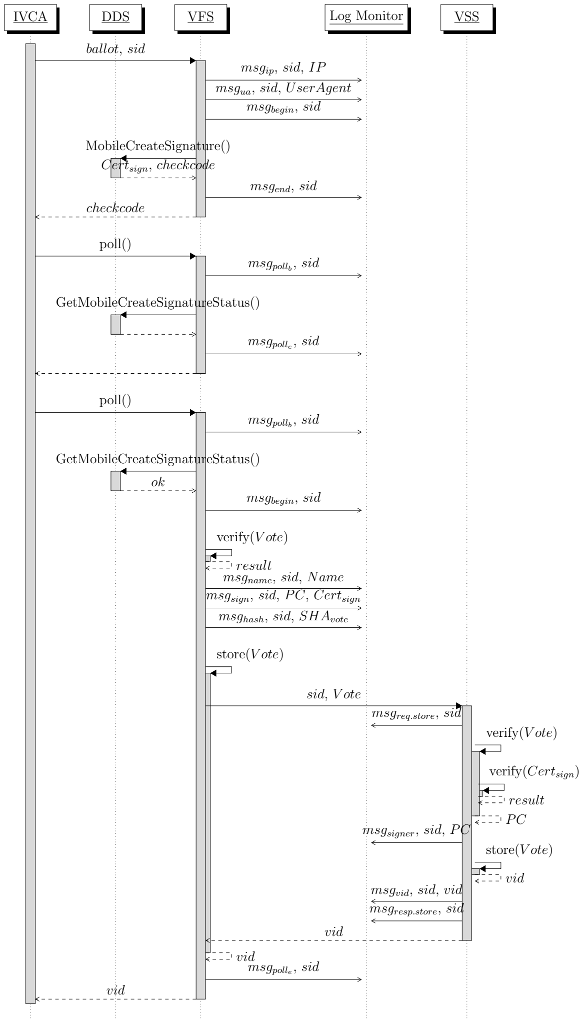


Figure 48: Logs generated on vote submission (Mobile-ID)

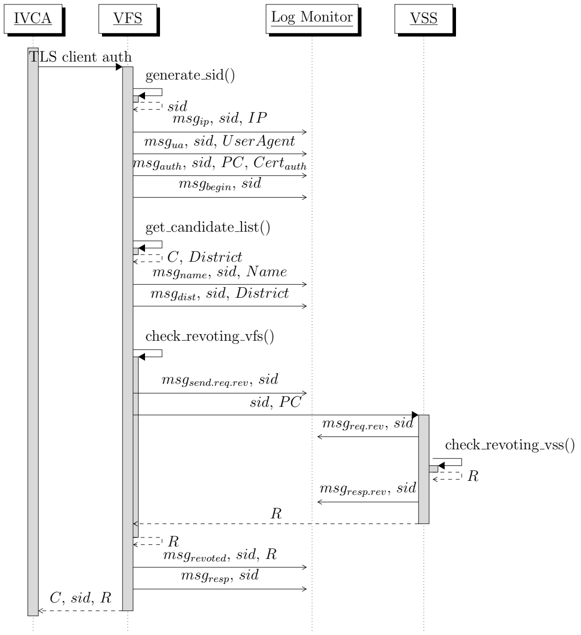


Figure 49: Logs generated on candidate list retrieval (ID card)

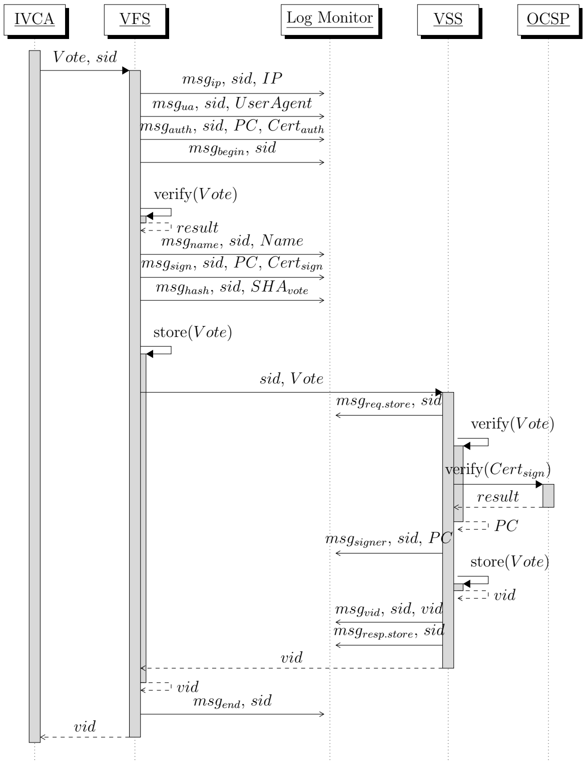


Figure 50: Logs generated on vote submission (ID card)

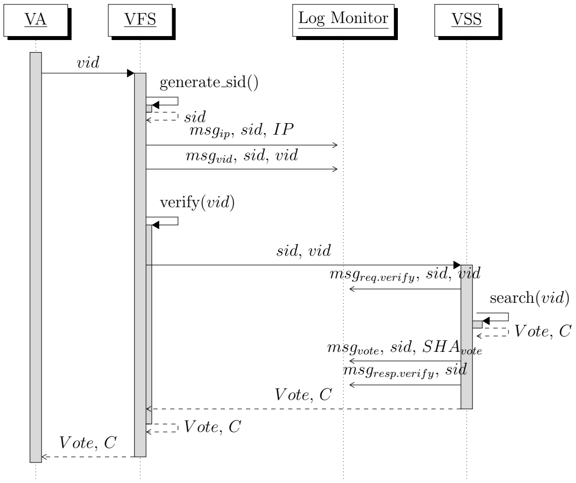


Figure 51: Logs generated on verification