Log Analysis of Estonian Internet Voting 2013–2014

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

[1] Smartmatic-Cybernetica Centre of Excellence for Internet Voting, Ülikooli 2, Tartu, Estonia
[2] Software Technology and Applications Competence Centre, Ülikooli 2, Tartu, Estonia
[3] Institute of Computer Science, Tartu University, J. Liivi 2, Tartu, Estonia
[4] Cybernetica, Ülikooli 2, Tartu, Estonia
jan.willemson@gmail.com

Abstract. This paper presents analysis of Internet voting system logs of 2013 local municipal and 2014 European Parliament elections in Estonia. We study both sociodemographic characteristics of voters and technical aspects of voting. Special attention is paid to voting and verification sessions that can be considered irregular (e.g. inability to cast a valid vote or failed verifications). We observe several interesting phenomena, e.g. that older people are generally faster Internet voters and that women use the vote verification option significantly less than men.

1 Introduction

The 2011 parliamentary elections were a significant landmark in Estonian i-voting. The share of votes cast over the Internet reached the as high as 24.3 % [7]. Such a share makes Internet voting an appealing target for various attackers, and indeed, several different attacks were observed in 2011 [9]. The most interesting one was proposed by a student who developed a proof-of-concept vote-rigging malware that exploited the lack of a feedback channel in 2011 elections.

As a result of these events, Estonian National Electoral Committee (NEC) took the initiative to improve the security of Internet voting in various ways. The i-voting protocol was extended by adding a new scheme providing cast-as-intended verification for Estonian i-voting [11]. A separate effort was established to perform in-depth analysis of logs produced by i-voting servers in order to study voter behaviour and to detect attacks against i-voting system and system malfunction.

This paper presents the results of these analysis efforts for 2013 and 2014 Estonian elections. Internet voting in 2013 local municipal elections (KOV2013) took place from 2013-10-10 09:00 to 2013-10-16 18:00 [5], and Internet voting in 2014 European Parliament elections (EP2014) took place from 2014-05-15 09:00 to 2014-05-21 18:00 [8].

To facilitate this kind of analysis, NEC has taken a decision to provide pseudonymised log records for research purposes. During pseudonymisation, the

R. Haenni et al. (Eds.): VoteID 2015, LNCS 9269, pp. 19–34, 2015.
DOI: 10.1007/978-3-319-22270-7_2

voters' identities and client certificates have been replaced by pseudonyms (leaving the option of studying repeating voting patterns). Sociodemographic data (gender, age) and technical session data (e.g. time stamps, OS identifiers) have been preserved in the logs in their original form.

There have been several reports on electronic voting log monitoring, but they have mostly been concerned with voting machines. Antonyan *et al.* analyse AccuVote optical scanning terminal logs [2]. Peisert *et al.* discuss the need for a detailed forensic audit trail to enable auditors to analyze the actions of e-voting systems [14]. Michel *et al.* present a grammar-based log analysis framework automating the analysis of event logs recorded by the electronic voting tabulators [12]. To the best of our knowledge, the current paper presents the first analysis of remote voting system logs.

The paper has the following structure. Section 2 describes the logging framework together with the criteria used to determine successful voting sessions. Sections 3 and 4 present various sociodemographic and technical metrics observed while studying the logs. Sections 5 and 6 analyse failed voting and verification sessions, respectively. Finally, Sect. 7 discusses the most interesting findings and makes some conclusions.

2 Log Monitoring for Estonian Internet Voting Scheme

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 digital signature (see [9] 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 2013 elections it is also possible to verify one's vote using a mobile device [11]. In case the Internet voter feels coerced, she can resubmit her vote via Internet or in the polling station during the advance voting period.

The three protocols implemented by the i-voting system – voting with smart card, voting with Mobile-ID and verification – are defined by finite-state machines. The transitions between the states generate log messages. For example Fig. 1 displays the protocol for candidate list retrieval with a smart card-based eID tool. After TLS authentication to 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 eligibility of the voter, checking the re-voting status at Vote Storage Server (VSS) and returning the candidate list to IVCA. Each of those steps is logged accordingly. The candidate

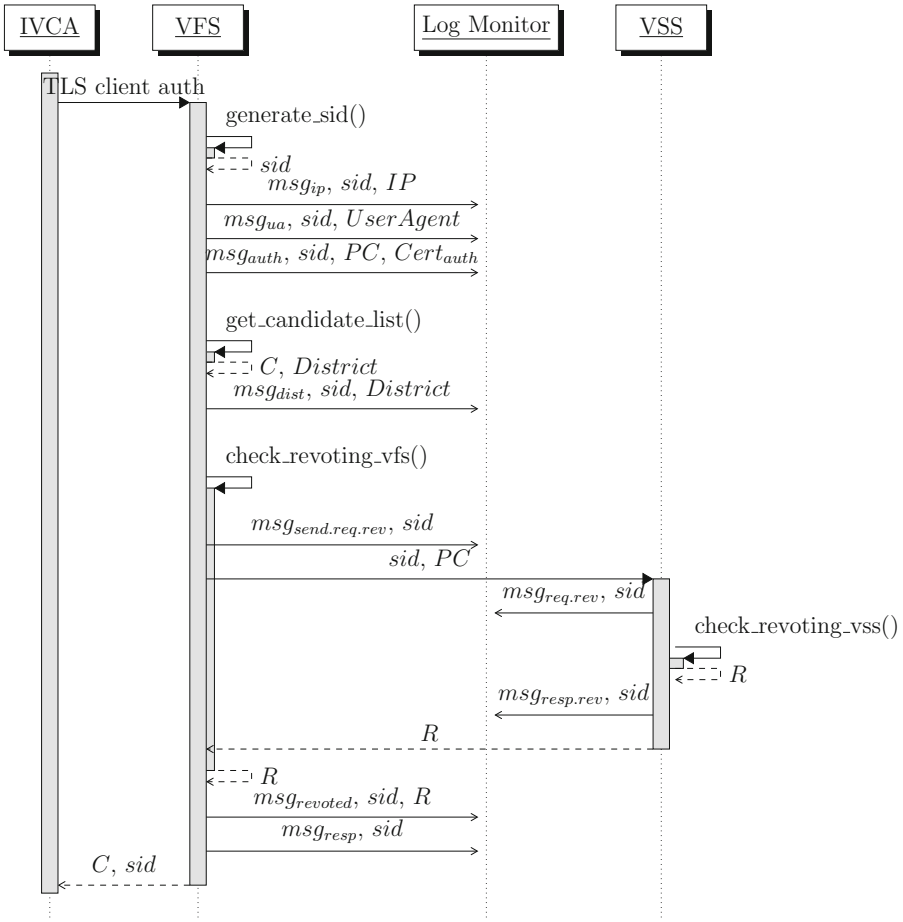

Fig. 1. Logs generated on candidate list retrieval

list retrieval is later followed by the vote casting where the same *sid* is submitted by the IVCA.

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 upon it. For example, unusually high system load could signal a possible bug in server software or an ongoing denial-of-service attack. Sudden increase in the number of unfinished voting sessions could be caused by a bug in i-voting software or an attack being performed on Internet voters, etc.

A log monitor has been introduced to the architecture. The monitor is connected to VFS and VSS receiving copies of log messages in quasi real-time using

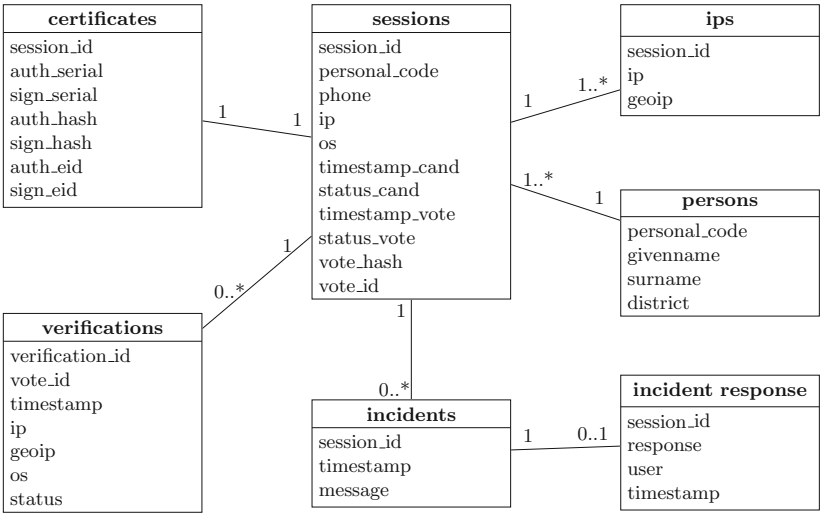

Fig. 2. Database table structure

`rsyslog` utility with UDP as transport protocol. The log monitor parses every log line, and by using regular expressions tries to match the line against the list of defined patterns. Useful information is extracted from the log entries 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.

Database table structure is shown in Fig. 2. The central table is `sessions` containing the data describing the voting session. The `verifications` table contains information about vote verification requests which 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 `incident_response` table, which stores incident resolutions created by election officials.

2.2 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 characterize legitimate program behaviours. Sessions that describe valid protocol runs and end with a successful result or acknowledged error state are generally not interesting for detailed analysis. These sessions are white-listed, they may become the subject of analysis in case some external condition characterizes them as a 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 the sessions. In case certain conditions are not met the

session is labelled for further analysis. The main criteria to call a voting session normal are given below.

1. *The encrypted vote is signed with the same eID tool that was used for authentication.* The i-voting protocol allows, e.g., for the voter to authenticate using ID card and submit a vote signed by Mobile-ID eID tool. However, that would be an anomaly since the official IVCA does not implement such a feature and there is no clear reason why the voter would use two eID tools in one voting session.
2. *The IP address and the OS of the voter do not change through the voting session.* Voter's IP address or OS change in the middle of the voting session might indicate voting session hijacking. Although we do not see the benefit or the flaw that would allow to hijack the i-voting session, we believe that detection of such an anomaly is advisable.
   Note that an IP address change could happen also for a completely legitimate reason, such as voter switching Internet connections in the process of i-voting.
3. *The vote cryptogram is unique.* To encrypt the vote, RSA-OAEP encryption scheme with random padding is used. Therefore, there should be no duplicate votes received by the i-voting servers. Several encrypted votes sharing the same cryptogram could indicate either randomness failure in IVCA (as was the case in 2013 parliamentary elections in Norway [3]), vote manipulation malware that uses hard-coded version of encrypted vote, or a ballot copying attack [4].
4. *Verification is requested only for those vote identifiers that have been issued.* Verification request containing a vote identifier which has not been issued by the i-voting server could indicate a vote identifier brute-force attack or a bug in the i-voting system or verification software. This event may also trigger legitimately if the voter is for some reason using a QR code from a previous election.

In addition to labelling sessions as normal or anomalous we also aggregate descriptive metrics about ongoing election. The gathered data contains sociodemographic metrics such as age and gender, technical data such as OS, eID tool, IP-addresses, etc. Some metrics are described below.

1. *Amount of voters sharing an IP address.* Several voters using the same IP address could indicate that a collective voting is being performed or the votes are cast by a single person who is using eID tools of other persons. However, several voters can be legitimately using a single IP if they are voting from a large organization where shared connection is used to access the Internet.
2. *The overall percentage of revoters.* In order to prevent vote selling and coercion, voters can change their i-vote by casting another i-vote. Through previous elections in 2005–2011 the revoter proportion has been between 1.15 % and 3.9 % [7]. A sudden increase in revoter proportion should be considered an anomaly indicating a large scale coercion or malware installed on the voting devices that revotes using voter's eID tool connected to the device, thus escaping detection by vote verification scheme [11, Section 5.E].

Increase in revoter ratio could also have a legitimate reason, e.g. a significant political scandal occurring during the voting period.
3. *Number of IP addresses for verifying a single vote.* By design, the vote verification protocol allows anyone knowing the vote reference to download the encrypted vote from the server. Under normal circumstances, we should see the vote verified from only a few devices (mostly just one). Verification requests coming from different devices may be an indication of the QR code containing the vote reference being misused.
4. *Amount of verifiers sharing an IP address.* Large number of votes verified from a single IP address may indicate a large-scale vote-buying attempt. On the other hand, verification from the same IP address can also happen if one mobile device or Internet connection is shared by several verifiers legitimately or a dynamic IP address is reassigned to different mobile devices.

3 Sociodemographic Metrics in 2013–2014

In this Section we will summarize some of the more interesting findings we observed from the vote session logs w.r.t. sociodemographic metrics (age and gender). In 2013, 170 804 voting sessions were made. In total, 133 808 voters cast at least one successful i-vote and 4542 (3.39 %) of them attempted to verify their vote. In 2014, 114 799 voting sessions were made. In total, 103 151 voters cast at least one successful i-vote and 4250 (4.12 %) of them attempted to verify their vote.

3.1 Age Distribution

The youngest person who (unsuccessfully) attempted i-voting in 2013 was 3 years old (in 2014, 7), and the oldest i-voter was 102 (in 2014, 103). The youngest vote verifier was 18 (in 2014 also 18) and the oldest was 97 (in 2014, 93).

The activity by age of voters (expressed as a percentage of all the eligible voters) and verifiers (expressed as a percentage of all the voters) are shown in Figs. 3 and 4 for 2013, and in Figs. 5 and 6 for 2014. We see that the most active voters are people of age 30–40.

An interesting phenomenon was observed when studying the relationship between the voter's age and voting session length (which is defined as the time between downloading the candidate list and submitting the vote). It turns out that contrary to what one might expect, older people are faster i-voters. The phenomenon is illustrated in Fig. 7 for 2013 and in Fig. 8 for 2014. We note that this phenomenon does not disappear when splitting the data by gender or eID tool used. The cause of this phenomenon remains unclear, possible reasons include older people having made up their minds already when starting to vote and younger people being more affected by multitasking.

Figures 9 and 10 show the histogram of general voting session lengths observed in 2013 and 2014 respectively.

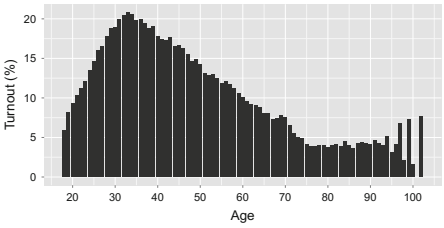

Fig. 3. KOV2013: Voter activity by age

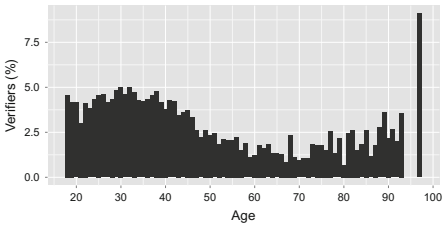

Fig. 4. KOV2013: Verifier activity by age

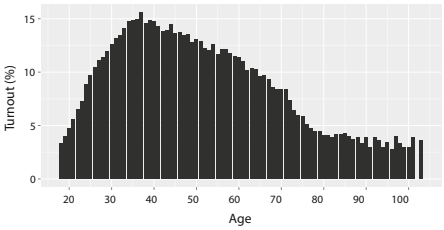

Fig. 5. EP2014: Voter activity by age

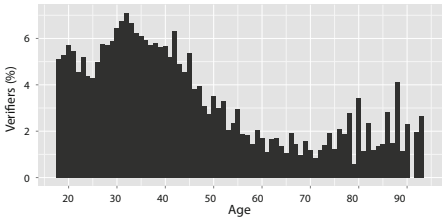

Fig. 6. EP2014: Verifier activity by age

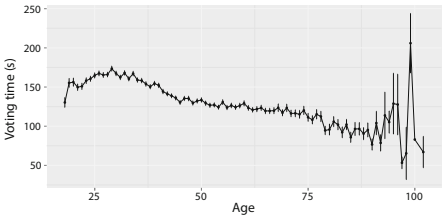

Fig. 7. KOV2013: Age vs voting time

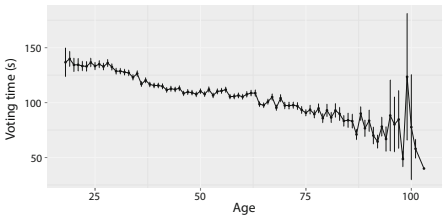

Fig. 8. EP2014: Age vs voting time

Table 1 gives 0.5 %, 1 %, 99 % and 99.5 % quantiles on the length of the voting session. It allows us to estimate that a normal length for a voting session could be considered between 20 s and 20 min (in 2014, 20 s and 13 min). Note that for 91.28 % (in 2014, 96.11 %) voting sessions the session length was less than 6 min.

3.2 Verification

It has been observed several times that women cast more i-votes than men. This observation was also confirmed in 2013 and 2014 elections when 52,2 % and 51.53 % of successful Internet voters were women, respectively.

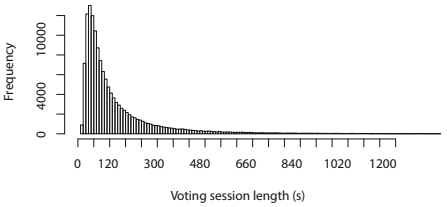

Fig. 9. KOV2013: Distribution of the voting session lengths

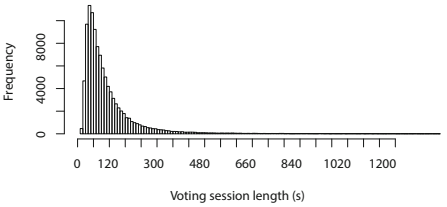

Fig. 10. EP2014: Distribution of the voting session lengths

Table 1. Quantiles of voting session lengths in seconds

Quantile	0.5 %	1 %	99 %	99.5 %
KOV2013	20	22	1182	1685.4
EP2014	21	23	751	1080

However, the gender distribution of verification is completely different, as only 31.6 % and 26.35 % of vote verifiers were women in 2013 and 2014, respectively.

It is also interesting to look at the length distribution of the verification operation (i.e. the time between the vote identifier has been issued and the vote verification request has been received). The period during which the server replies to the verification request with the vote cryptogram has been limited to 30 and 60 minutes in 2013 and 2014, respectively. Several verification requests were still received significantly after the end of this period. For example, in 2013, 19 voters made their first verification request 1 hour after the vote had been submitted, 10 voters did it 6 hours and 6 voters 1 day after the vote submission.

Frequencies of verification lengths (taking into account only the first verification request made by the voter) are shown in Figs. 11 and 12 for 2013 and 2014, respectively.

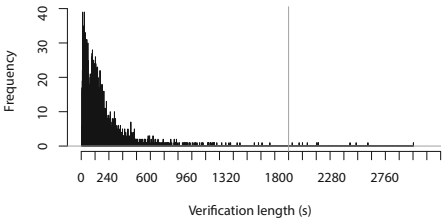

Fig. 11. KOV2013: Distribution of verification lengths

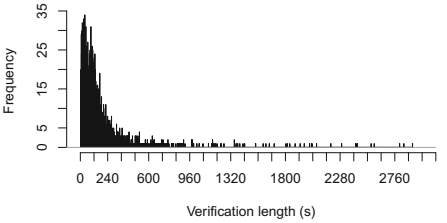

Fig. 12. EP2014: Distribution of verification lengths

4 Technical Metrics in 2013–2014

4.1 OS and eID Distribution

The official IVCA is available for three operating system families. Table 2 shows the popularity of Windows, Mac OS X and GNU Linux for voting.

As mentioned in Sect. 2.1, an i-vote can be cast using three eID tools. Popularity of these tools is shown in Table 3. Note that in Tables 2 and 3 we only take the final votes into account, thus excluding the votes annulled by revoting.

Table 2. OS distribution

OS	Windows	Mac OS X	GNU Linux
KOV2013	93.87 %	5.35 %	0.78 %
EP2014	93.4 %	5.46 %	1.14 %

Table 3. eID distribution

eID	ID card	Mobile-ID	Digi-ID
KOV2013	90.27 %	8.49 %	1.23 %
EP2014	87.69 %	10.86 %	1.45 %

4.2 IP Address Shared by Several Voters

In 2013, 133808 (in 2014, 103151) voters used 68503 (in 2014, 52191) unique IP addresses to cast their successful votes.

There were 28 (in 2014, 22) IP addresses which were each shared by more than 100 voters with the top IP address shared by 1127 (in 2014, 970) voters. We reviewed the top shared voting IP addresses 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. This is consistent with the expected behaviour of people voting from one large organisation having just one external IP address.

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, e.g. voters' casting their votes shortly after each other. Table 4 shows the number of voter groups observed, where voters voting in 5 min interval between each other and using the same OS are considered to belong to one group.[1] The table contains data only about those IP addresses which do no have overlapping voting sessions and those with the first and last voting activity falling into a 24 hour window.

4.3 Revoting

In 2013, 1.93 % (in 2014, 1.69 %) voters (altogether 2586; in 2014, 1743) cast more than one vote. From these revoters majority revoted only once. It appears that 30 % (in 2014, 28 %) of the revoters revote in the first 10 min, and 41 % (in 2014, 38 %) of revoters revote in the first hour after casting the vote.

[1] This definition of a group is limited to phenomena observable from system logs. A proper group voting study would require a more detailed social science approach.

Table 4. Voter groups in 2013 and 2014

Group size	KOV2013	EP2014
2	8476	6033
3	697	523
4	108	60
5	15	9
6	3	1

Figures 13 and 14 show distribution of votes and revotes over the voting period. We see that revoting activity pattern over the voting period follows the voting activity pattern.

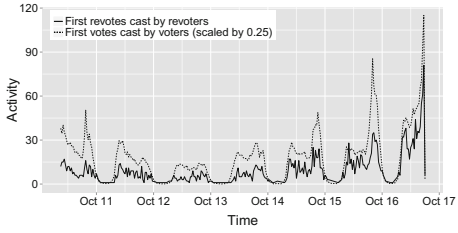

Fig. 13. KOV2013: Distribution of votes and revotes

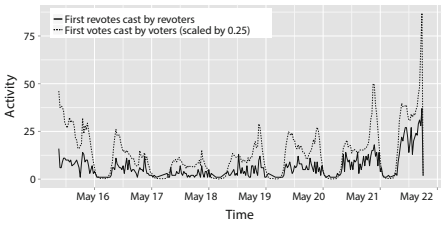

Fig. 14. EP2014: Distribution of votes and revotes

We can estimate that in the worst case in KOV2013 2586 (in 2014, 1743) voters could have been coerced or fallen as a victims for revoting malware described in Sect. 2.2.

However, since in the previous elections revoter proportion was similar (see Sect. 2.2) and some amount of revoters is normal, it is unlikely that most of the revotes would have been caused by an attack.

5 Unsuccessful Voting Sessions

From those persons who attempted to i-vote in 2013, 96.6 % (in 2014, 98.5 %) succeeded to cast at least one successful vote (possibly by retrying). Still in 2013, 19.88 % of the voting sessions (in 2014, 8.39 %) did not result in a successfully cast vote. In this section we present the causes for unsuccessful voting sessions.

5.1 Sessions Failing with an Error Condition

It is natural for some voting sessions to fail – e.g. it is possible that a person is not in the list of eligible voters by mistake and finds it out only during the

failed attempt to 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)") are given in Table 5.

Looking at the row "Ineligible voters", we can see that in 2013 some voters who were originally declared ineligible eventually still managed to submit a vote. This is because a person's eligibility status can change during the voting period.

Table 5. Failed voting sessions in KOV2013 and EP2014

	KOV2013			EP2014		
Reason of failure	Sessions	Voters	Voters (u)	Sessions	Voters	Voters (u)
Maintenance of voting servers	11	11	1	–	–	–
Under-aged voter	28	25	25	16	16	15
Ineligible voter	1063	774	766	315	199	199
Voting not started	3	2	0	5	3	0
Voting already ended	1	1	1	38	35	28
Pre-2011 Mobile-ID user	1490	876	332	549	407	160
Bad Mobile-ID number	2051	–	–	491	–	–
Mobile-ID failure (auth)	2004	1394	122	1200	776	54
Mobile-ID failure (sign)	1043	926	29	609	562	33
Revoked ID card	1933	872	755	270	146	128
Revoked Mobile-ID	41	–	–	23	–	–
Incident	93	60	6	1173	325	88

5.2 Failure to Cast a Vote

Some voting sessions did not fail because of an error, but were from the i-voting system perspective simply abandoned – the candidate list was successfully downloaded, but the vote submission request did not follow. Table 6 shows the number of affected voters and the number of voters who did not manage to cast any i-vote.

From these 2889 (in 2014, 869) voters, 176 (in 2014, 20) voters had at least one voting session with failed status. From the remaining 2713 (in 2014, 849) voters, 2000 (in 2014, 700) voters had made only a single voting session which did not continue after candidate list retrieval, 370 (in 2014, 79) voters had two such sessions, 52 (in 2014, 9) voters had more than six such sessions.

Some of these unfinished voting sessions in KOV2013 can be explained by a bug [1] in libcurl library used by the IVCA which caused a connection timeout when sending vote submission request over a slow network connection.

We can only speculate why these voters did not get past the candidate list retrieval in EP2014. Possible reasons include forgetting the PIN required to sign a vote with an eID tool, not finding a suitable candidate in the downloaded candidate list, or simply not realising that the i-voting session has to be completed by signing the vote.

5.3 Incidents

In addition to previously defined error conditions and abandoned voting sessions, there were 93 (in 2014, 1178) voting sessions raising an incident alert caused by unexpected log entries.

KOV2013. We observed 37 ID card voting sessions failing with the incident message which stated that the signing certificate digest did not match the digest specified in the vote. In total 12 voters were affected. Almost all of the voters were using GNU Linux OS except one voter who was using Windows OS. The incident was traced to the bug in OpenSC smart card library which was shipped with some GNU Linux distributions [13]. The bug failed to remove zero padding from the certificate when reading it from the smart card.

On 2013-10-15 from 15:12:26 to 15:13:08 there were 36 failed voting sessions logged with an incident message which informed about unavailability of VSS. The reason for VSS downtime was vote backup routine which required to stop Apache process running on VSS. Starting from the next elections (EP2014) LVM snapshots were used which allow to backup the votes without stopping the Apache process.

We observed 17 incidents caused by malformed votes – some votes were rejected as invalid. Altogether 13 voters were affected, all of them later managed to successfully cast an i-vote. Some of these incidents were traced back to the bug in IVCA. The IVCA continued with vote submission even if the certificate could not be read from the smart card or the digital signature generation in the smart card failed. In case of one voter it was found that the failure was caused by a defective Mobile-ID SIM card. Without the corresponding invalid votes, some of those incidents could not be thoroughly investigated.

We observed 3 Mobile-ID voting sessions which raised an incident alert about invalid phone number received. The problem was traced back to IVCA which failed to correctly enforce valid phone number input from the voter.

EP2014. We observed 1131 voting sessions failing with an error message stating that the certificate used to sign the vote is not yet valid. The error was traced back to a bug in server-side software updated in EP2014, which did not take into account timezone information when checking the certificate validity date. The error affected voters who had renewed their eID tool on the day of i-voting. The voters who approached NEC support centre were instructed to retry after a

Table 6. Abandoned voting sessions in KOV2013 and EP2014

	KOV2013			EP2014		
	Sessions	Voters	Voters (u)	Sessions	Voters	Voters (u)
Sessions without cast votes	24103	15563	2889	4921	3889	869

few hours. From the 310 voters affected, 229 managed to successfully cast their i-vote later in the i-voting period.

We observed five ID card voting sessions where the person submitting the vote was not the same who obtained the candidate list. The behaviour can be explained by the new "Retry" button feature introduced in IVCA which allows to obtain the candidate list using one ID card, but sign and submit the vote with another by swapping the cards between these operations. These votes were accepted and counted without creating a problem. While it is not the case in European Parliament Elections, it may happen that the voter obtaining the candidate list and the voter casting the vote have 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 other person's ID card before signing the vote. The persons involved in these sessions were paired as 79 years old male and 72 years old female, 50 years old male and 71 years old female, 74 years old male and 68 years old female, 56 years old male and 58 years old female, and 52 years old male and 33 years old female. From the voters who obtained the candidate list, two submitted their own vote a few minutes later, but three voters did not cast their vote at all.

We observed one ID card voting session using Windows IVCA failing with the incident alert stating that the vote signature could not be verified. 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 the currently valid ID card before signing the vote with an older ID card which had been officially reported lost.

The rest of the incidents were caused by the bug in a smart card library or person retrying the failed Mobile-ID session.

6 Unsuccessful Verification Sessions

6.1 Failure to Verify

In 2013, NEC received no complaints about unsuccessful vote verifications. However, we see that for 33 (in 2014, 26) voters their first verification attempt was not successful, resulting in an error message shown to the voter. Those voters tried to verify after the verification time-window had passed or after a new vote was submitted by them. In 2013, one verifier failed because VSS was unreachable due to backup procedures.

In 2014, we observed 196 vote verification requests having a malformed vote ID. Some of malformed vote ID requests were caused by a bug in iOS-based vote verification application which truncated the vote identifiers that contained a 0-byte. Four voters called to NEC support centre complaining about iOS verification application being unable to find their vote on VSS. The bug was fixed

during elections and updated iOS application was pushed in iOS app store [11, Sect. 6].

However, other malformed verification requests could not be attributed to 0-byte bug. The malformed vote verification requests were traced back to iOS vote verification application, which failed to validate contents of the captured QR code before forming the vote verification request sent to VFS. This bug in iOS-based vote verification application has been fixed.

6.2 Verification Requests that Could Not Be Linked to Votes

We observed vote verification requests for three (in 2014, five) unique vote identifiers that were not issued by i-voting system. Some of those vote identifiers were queried multiple times by several unique IP addresses. One of the identifiers seen in EP2014 was also seen in KOV2013. We were able to track one of those identifiers to a QR code from information materials about Internet voting.

7 Discussion and Conclusions

7.1 Summary of the Findings

Log monitoring has proven to be a useful tool for the election officials for troubleshooting voters' problems and understanding the state of ongoing i-voting. In KOV2013 and EP2014 several malfunctions in IVCA, i-voting system, verification apps and external systems were discovered and fixed. From the i-voting perspective, those bugs were causing minor inconveniences to voters, in most cases it was possible to re-vote successfully.

In those elections we did not observe any event which could qualify as an attack against i-voting system. Furthermore, taking into account all observations, we can conclude that during KOV2013 and EP2014 no large-scale attack has been executed against i-voters.

There were several interesting phenomena observed in the logs that were unknown before. We were able to determine that older people are generally faster i-voters and vote verifiers are predominantly men, even though among the general population of i-voters the share of women is slightly higher.

7.2 Limitations of the Approach

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

In some of these cases the causes might be found if the voter could be contacted for an explanation. However, there is no simple way to contact the voter[2] and there is no legal basis for it, unless there is convincing evidence that illegal

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

activity might have been performed. The only event when the voter was contacted, was the case of voter who cast more than 500 votes in RK2011 [10], 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 unavailability of the vote involved in an incident. Logging and availability of such data for investigation is deliberately limited by NEC due to the vote privacy concerns.

Obviously, the approach used in this work can detect only the attacks executed by external attackers who attack voters' voting devices or eID tools, since none of the anomalous patterns applied 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.

After the i-voting server-side source code was published on GitHub [6], the described 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 his attacks on a live election system. The exploit can be developed using a cloned i-voting system fully operated by the attacker.

Note that Internet voting still has a significant human component and hence not all the errors can be expected to manifest themselves only on digital media. For example, the mobile application for vote verification only displays the candidate number found in the cryptogram, but the decision about its match with the voter's intention is taken inside her head. Thus, some parts of the analysis of events depends on the voters' willingness to report them.

Also note that most of the reasons for suspicion do not necessarily indicate a malicious attack and can occur for perfectly acceptable reasons. However, they can be a starting point for more in-depth analysis to draw more complex conclusions (e.g. in case several of the items trigger a flag).

7.3 Future Work

Most of the anomalous patterns – e.g. IP address changing in the middle of the voting session, voter revoting several times – are not easily distinguishable from the legitimate behaviour. In some of those cases sessions become interesting only if a certain threshold is reached. Setting threshold values is a delicate trade-off between missing an attack and getting too many false positives. The statistical model of the expected behaviour built from KOV2013 and EP2014 data can be used to implement better anomaly detection for further elections. However, human behaviour is ever-changing, so these kinds of log monitoring efforts must be continued to adjust the normality profiles in the future accordingly.

Acknowledgements. This research was supported by the Estonian Research Council under Institutional Research Grant IUT27-1, 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. Timeout for Expect: 100-continue as an option, Oct 2013, Curl-library mailing list archives. http://curl.haxx.se/mail/lib-2013-10/0142.html
2. Antonyan, T., Davtyan, S., Kentros, S., Kiayias, A., Michel, L., Nicolaou, N., Russell, A., Shvartsman, A.: Automating voting terminal event log analysis. In: Electronic Voting Technology Workshop/Workshop on Trustworthy Elections (EVT/WOTE09) (2009)
3. Bull, C., Nore, H.: Problems encountered. Seminar on Internet voting, Sep 2013. https://www.regjeringen.no/contentassets/c41c2959b8d946bf8007b546552ff9dc/5_problems_encountered.pdf
4. Cortier, V., Smyth, B.: Attacking and fixing Helios: an analysis of ballot secrecy. J. Comput. Secur. **21**(1), 89–148 (2013)
5. Estonian National Electoral Committee: Municipal Elections 2013 Results (2013). http://kov2013.vvk.ee/
6. Estonian National Electoral Committee: Source code of the server side components of Estonian internet-voting system, Jul 2013. https://github.com/vvk-ehk/evalimine
7. Estonian National Electoral Committee: Statistics about Internet Voting in Estonia (2013). http://vvk.ee/voting-methods-in-estonia/engindex/statistics
8. Estonian National Electoral Committee: European Parliament Elections 2014 Results (2014). http://ep2014.vvk.ee/detailed-en.html
9. Heiberg, S., Laud, P., Willemson, J.: The application of I-voting for estonian parliamentary elections of 2011. In: Kiayias, A., Lipmaa, H. (eds.) VoteID 2011. LNCS, vol. 7187, pp. 208–223. Springer, Heidelberg (2012)
10. Heiberg, S., Willemson, J.: Modeling threats of a voting method. In: Zissis, D., Lekkas, D. (eds.) Design, Development, and Use of Secure Electronic Voting Systems, pp. 128–148. IGI Global, Hershey (2014)
11. Heiberg, S., Willemson, J.: Verifiable internet voting in Estonia. In: Krimmer, R., Volkamer, M. (eds.) 6th International Conference on Electronic Voting 2014, (EVOTE 2014), 28–31 October 2014, Bregenz, Austria, pp. 23–29. TUT Press, Tallinn (2014)
12. Michel, L.D., Shvartsman, A.A., Volgushev, N.: A systematic approach to analyzing voting terminal event logs. USENIX J. Election Technol. Syst. (JETS) 2(2), April 2014. https://www.usenix.org/system/files/jets/issues/0202/overview/jets_0202-michel.pdf
13. OpenSC project: Regression in e35febe: compute cert length, Dec 2012. https://github.com/OpenSC/OpenSC/pull/114
14. Peisert, S., Bishop, M., Yasinsac, A.: Vote selling, voter anonymity, and forensic logging of electronic voting machines. In: 42nd Hawaii International Conference on System Sciences, 2009. HICSS 2009, pp. 1–10. IEEE (2009)

http://www.springer.com/978-3-319-22269-1

E-Voting and Identity
5th International Conference, VoteID 2015, Bern, Switzerland, September 2-4, 2015, Proceedings
Haenni, R.; Koenig, R.E.; Wikström, D. (Eds.)
2015, IX, 173 p. 33 illus., Softcover
ISBN: 978-3-319-22269-1