On the Estonian Internet Voting System IVXV

Shymaa M. Arafat

Historical & Political Brief

Estonia is a small 1.9m citizen country located in east Europe who gained independence from Russia in 1991 and joined the European union in [1]. Most Estonian citizens welcomed the general digital transition in 2001; however, when it came to e-voting in 2005 there were some kind of “notable divisions within the society between those who fully trust and those who fully distrust internet voting” as quoted from the OSCE ODIHR June 2023 report [2].

You can trace a long history of rejection incidences from conservatives right parties in [1,2] from 2005 up till now; the situation was emphasized in 2023 when the internet votes flipped the election results for one of those parties, EKRE. The distribution of internet votes was completely different than that for poll station votes as shown in the curves in [3]; analysts due that to the society division mentioned above (ie., it’s expected for the distribution to reflect the parties conventions on the voting method), but still there are some complaining movements like [4]. Another technical incidence that gained some publicity [5] was done by the same computer scientist observer in [3]; he voted using his own code [6,7], meaning that he has overridden the official voting application (we will detail that in the next section); recently there were a mention of some wrong district votes too [8], I think the OSCE report said they were corrected except one vote.

Finally, I don’t think one can neglect the effect of Ukrainian war on Estonia being the closest neighbor. I have only read this article [9]; even if it was politically biased I think it points out to the economic status with more refugees that the writer projects will seek Estonian citizenship, and thus can vote. Given footnote in the OSCE report which says that Estonia is not very strict in giving citizenship for stable residents.

A Brief on Technical Evolution

-Since 2001 Estonia started a plan to digitize every thing in the country from e-government to the private sector, you can find the details of the remarkable status quo like Xroad, KSI private blockchain,… in [10]; e-ID cards exist since 2002, Fig.1, and electronic transactions is
is the casual behavior of the Estonian citizen.

Fig.1: Estonian eID card with 2 keys (authentication & signature), image taken from []

-Internet Voting first attempt was in 2005 and evolved till the last election 2023 with an increase in the number of voters from to as from the official statistics site [11]. It's worth mentioning that Tartu University always played a consulting role in the process with several reports and requirements; you can find one of the most MOOC e-voting courses there that I unfortunately missed last October [12].

-We will go through the system evolving details as attacks and/or vulnerabilities were discovered in the following section, but let me emphasize on one major event that changed a core cryptographic component of the e-ID system, **RSA**, before we look in depth to the details of the current design:

In May 2018, Estonian authorities officially declared a persisting problem that started to appear in some rare incidences of duplicate RSA keys since 2011/2012; Table1.

Certificate pairs with duplicate RSA public keys

No	Time of cert issuance	Type	Cardholder	Issuance	Expiry date	Revoked	Warranty
1	2012-11-06 15:35:09	sign	Ulle	PPA renewal	2016-07-07	2016-06-27	2014-10-09
	2012-11-06 15:35:46	auth	Toivo	PPA renewal	2016-07-04	2014-11-21	2014-10-09
2	2013-02-06 15:35:54	auth	Phillip	PPA renewal	2016-11-14	2015-05-04	2015-01-06
	2013-02-06 15:35:56	sign					
3	2013-02-07 12:18:34	auth	Sandra	PPA renewal	2016-01-02	expired	not issued
	2013-02-07 12:18:37	sign					
4	2013-02-19 09:09:58	auth	Nadiia	PPA renewal	2016-11-24	2016-11-08	2014-12-22
	2013-02-19 09:10:08	sign					
5	2013-02-25 09:33:17	auth	Moonika	PPA renewal	2016-08-22	2014-12-30	2014-12-22
	2013-02-25 09:33:29	sign					
6	2013-03-04 11:36:08	sign	Richard	PPA renewal	2016-11-30	2014-10-13	2014-10-09
	2013-03-04 11:36:38	auth	Anu	PPA renewal	2016-08-12	2014-10-23	2014-10-09
7	2013-03-30 13:40:38	auth	Leili	initial	2018-03-26	2015-05-14	2014-12-22
	2013-03-30 13:40:40	sign					
8	2013-03-30 13:42:03	auth	Jaan	initial	2018-03-26	2014-12-30	2014-12-22
	2013-03-30 13:42:05	sign					
9	2013-04-15 09:16:11	auth	Liis	PPA renewal	2016-05-06	expired	2014-12-22
	2013-04-15 09:16:28	sign					
10	2014-10-08 12:01:16	auth	Siim	initial	2019-10-07	2017-10-03	not issued
	2014-10-08 12:04:31	sign					

Table1: First incidences of duplicate RSA keys that were told to renew their ID cards at PPA stations; the table adopted from [13] with a detailed analysis of the marked with red case (they proved a valid signature for the first using the keys of the second)

The incidences requiring citizens re-installing the Java Applet on the cards at PPA (the issuing authority) stations (otherwise the card transactions will be suspended after a certain time limit) became more frequent, and hence providing more data & information for researchers to analyze, until it was proven that the ID card manufacturing company, ***Gemalto***, generated the RSA keys outside the chip (could be to fasten the process) which violates the agreement rules and gives a chance for the key pairs to be copied and repeated. A lot of interesting details on how the

analysis was done can be found in the the presentation [13] and the paper itself [14]; more faulty keys issues[1] can be found in the author's PhD [15], and in another independent work [16]. The authors in [1] say it was a global crisis and the company was sued in many other countries around the world; anyways Estonia did not change just the company to ***IDEMIA*** [17], but also the public key algorithm [18] moving to ***384-bit*** *Elliptic Curve Cryptography* ***ECC*** *public key encryption*[2]*,* but still without replacing the physical cards like what happened in Spain and Slovakia for example [19].

I-voting IVXV

After covering the issues in the eID keys voters use to authenticate themselves and sign their votes, we now explain the design and structure of the Estonian internet voting system, IVXV, as described in the official documents [20]. (this statement could be in the last paragraph in introduction when I rewrite it as a paper)

-In short, the developing company is Cybernetica; the voting device must be a desktop PC mobile voting is still postponed to at least 2025 [21]; multiple voting is allowed to avoid coercion or vote buying (only last vote is counted and poll station voting removes i-voting); El-Gamal Homomorphic Encryption scheme is used to encrypt votes then the encrypted vote is digitally signed by the voter(double envelope); optional vote verification can be done by voters (through QR code[3] using a second device) within 30 mins of voting with a max of 3 times; Mixnets are used to scramble votes before decryption to preserve ballot secrecy.

The system architecture and voting steps are depicted in Fig.2; we can summarize the steps (skipping the cryptographic details and equations for now)

-The voter installs the voting application[4] on his/her PC

-After submitting eID (or mobile ID), the voting application checks the eligibility of the voter to vote (through the registration application) and if eligible displays the candidate choices for that

---

1 Example errors include codes printed too dark which made them readable using torch, without opening envelope (happened twice in 2002 with the old company then again in 2018: https://news.err.ee/886313/new-id-card-issue-codes-can-be-read-using-torch-without-opening-envelope), duplicate email addresses in certificates, issuing certificates with incorrectly encoded public keys, failing to revoke certificates of deceased person's.

[2] a small note here is that according to [20] the "authorized voters list" is still signed using an 2048 bit RSA key

[3] According to [6], there were also a revealing incidence of the president vote through his QR code (when i-voted live in front of cameras, and showing his QR code, to encourage citizens to vote online; people took a snapshot of the QR code and revealed his vote). The incidence was mentioned in the context of doubting privacy and protection from coercion and/or vote buying.

[4] Sometimes called the voter application in official documents, but we prefer to follow the naming convention in [23] to make it clearly distinguishable from the voter.

voter (according to district if it's a local election)

-The voting application encrypts the voter choice using the election public key (El-Gamal encryption), then adds the user signature on the encrypted vote (with the voting application running on the voter's PC and after the voter's approval, the voting application has the right to sign a message with the voter signature), and also the timestamp certificate received from the registration application[5].

-After validating the voter's signature, the signature is removed and the encrypted vote is added to the list of votes (to be mixed then decrypted at the counting phase)

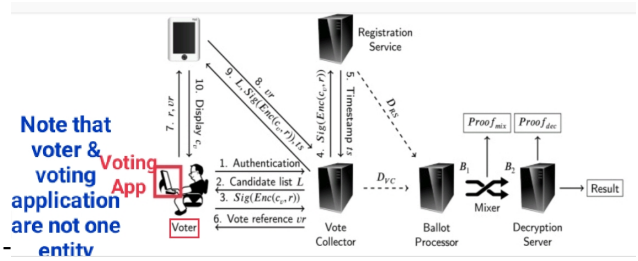

Fig.2: a diagram describing the architecture & the steps of the Estonian voting system, adopted from [1] which refers to [] as the original source; however, we remind the reader to distinguish between the voter machine and the voting application when it comes to attacks & vulnerabilities

-The introduction sections of [22,23] provide a condensed brief on the gradual historical improvement of the Estonian i-voting to fix and avoid attacks and/or vulnerabilities whenever they were discovered. So, let's highlight the last fixed attack in Feb 2023 just before the elections that were held in March[24, line in code][6]; then dive into the current system status with the remaining vulnerabilities.

Cryptographic Details

To understand the vulnerability, we have to go through the equations; we will also omit the voter signature and mixnets parts and concentrate on the part shown in Fig.3

[5] The timestamp certificate is important to distinguish the last vote of each voter and also for the valid verify duration of 30 mins.

[6] One may find it suspecious that they didn't fix it when the authors first sent them a letter and waited till the last month, while the paper was officially published before that with the authors stating that uptill now the vulnerability was not fixed and that whenever they asked the reply was "we're working on it".

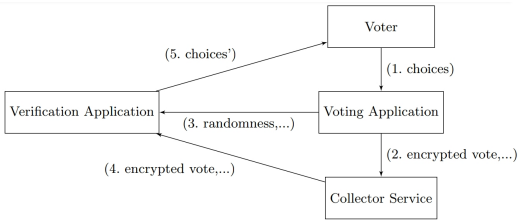

Fig.3: the part of vote casting and verification taken from [23]

Let's assume the election public key "y" with corresponding secret key "Sk", and a generator "g" for El-Gamal encryption, then to encrypt a vote "v" the voting application generates a random number "r", so that the encrypted vote is

$(C1, C2)= (g^r, y^r v)$

-The verification application working instantly within 30 mins receives "r" from the voting application (hidden in the QR code) and calculates

$v = C2/ y^r$ the voter is assured when the displayed "v" is the same "v" he/she voted for.

-When counting votes the election authority uses the election secret key (Sk) and the El-Gamal encryption known equation. $y= g^{Sk}$ to calculate $v= C2/(C1)^{Sk}$

In the older design the verification application received only C2; this gives a malicious voting application the chance to manipulate the encrypted cipher text by sending wrong "r" value to deceive the verification application (different values of C1 for the same C2) . Long story short, the authors found three possible manipulations with a simple fix of the vote collector to send the whole encrypted pair (C1,C2) to the verification application such that it also verifies that $C1= g^r$ as was finally done [24, lines 77-83 and the exception is thrown at line 60].

Remaining Vulnerabilities/Issues

-However, with ***verification ratio of 5.5%*** as stated in the official i-voting statistics site [11], this does not really prove beyond reasonable doubt that no encryption pairs were manipulated. The vulnerability here (and other vulnerabilities as well) comes from the possibility of a malicious voting application; [23] commented on the voting application not revealed as an open source, and the incident we mentioned at the beginning [6,7][7] proves that it is not even authenticated; the OSCE report [2] also notified about the risk of ***not authenticating the voting application***.

-This leaves it as an open challenge for adversaries to design the most possible malicious one with probability ~ 94% that the voter will not verify to detect it; a ratio that could even increase

[7] The one he developed was in Python, while IVXV code is written in Java

by social engineering to target those who are not likely to verify.

-Another possible risk is for ***vote buying applications*** to do what the authorities haven't done (authenticating their official voting application); ie, develop a fixed candidate voting application and authenticate its use through execution attestation[8] on the voter PC before transfering the money. This ***DarkDAOs*** idea was discovered by [25] in 2018 as a possible threat to decentralized voting in DAOs using governance tokens, but it could happen here too; the authors published a follow-up in 2023 [26] with a GitHub code.

-Another malicious voting application attack that could deceive even verifying voters was discovered by ***Olivier Pereira*** in [27]; the application could fake a system crash to take the voter signature twice (generate two votes and two "r" values) and deceive the voter to vote again while showing the QR code of his/her choice that the system will consider an old vote. Although the author suggested few mitigations, we have no clue that any of them was adopted. A simple trivial solution (for this specific problem) is to *force a time interval between votes*[9]; the verification interval, 30 mins, seems a suitable choice.

-A general solution to all the above would be *to* ***authenticate the official voting application either through checking its file digest (hash SHA256 for its code for example) or assigning to it a signature & authentication key pair*** *like the rest of involved applications in the system*. However, another issue remains of how to ***inform a non-verifying voter*** *that the vote was rejected because he/she has installed a malicious voting application*; for example the vote collector application could deduce the IP address of the voter machine from the first contacr with the voting application, then it can send a direct warning message to the voter screen, see [28] for some suggestions.

-Another problem that was mentioned in the OSCE report [2] is that there's no cryptographic proof for the deletion of multiple votes or ill-formed vote ballots; ie., the authorities are assumed trusted regarding not deleting or adding extra votes at this step. Quoting their own words "***The critical step of removing the votes overwritten by another vote cast*** *on the internet or in a polling station was not audited", "An insider with sufficient resources to alter the system, if able to do so undetected, could manage to control which votes are removed and therefore partially impact the results".* Meaning that yes there are decryption proofs that what got into the mixnets are what got out of it and finally decrypted, and there is the possibility to design a public

[8] Remote execution attests were originally discussed on Intel SGX (which is available on many new PCs in the market: https://www.intel.com/content/www/us/en/architecture-and-technology/software-guard-extensions-processors.html), and it exists in other processor companies like Apple and others as well. So, we can assume there's a considerable probability that the voter PC can support it.

[9] I also sent few suggestions to information systems emails from (https://www.valimised.ee/index.php/en/electoral-organizers/state-electoral-office/staff), and the time interval is the one they considered "possible" in their reply.

decryption proof verifier [22], but there is no cryptographic proof for the transition from the total list of votes to the "to be counted" list of votes.

-For this problem we suggest to aggregate all votes [28] in a 2-level Authenticated Data Structure ADS; if we use a ***Verkle Tree to aggregate all votes***, such that all ***votes from the same voter are aggregated in a second level Merkle Tree***[10], Fig.4.

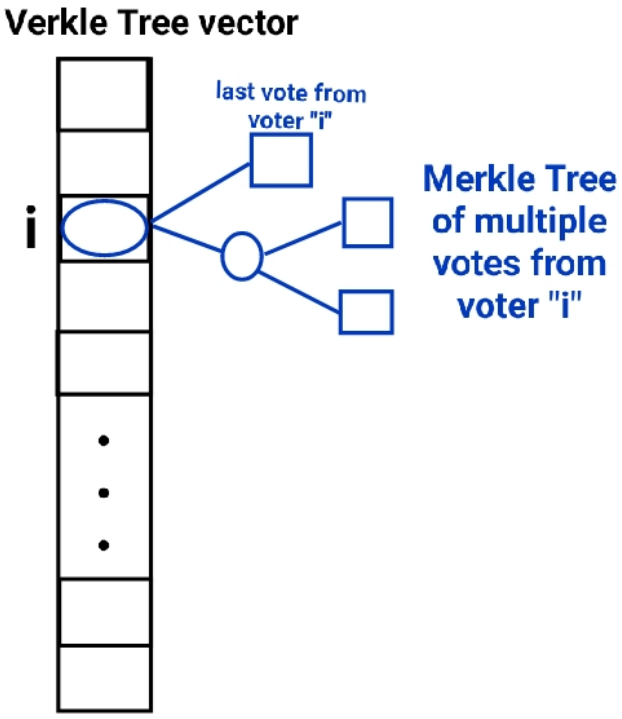

Fig.4: two-level Authenticated Data Structure, a vector that will be committed to using KZG commitments as a Verkle Tree, where each node value could be the root of a Merkle Tree containing multiple votes from the same voter

-When a new vote enters the system (through the vote collector and the registration applications):

*If the attached voter signature hasn't appear before, the new vote is added to the votes list and also aggregated to the Verkle Tree vector commitment.

*If it's a repeated signature, the new vote is inserted to the Merkle Tree attached to the corresponding Verkle node and then the Verkle node is replaced by the new Merkle root (the old root node is deleted and the new root is inserted).

[10] we use Merkle in the second level because we expect no voter will vote more than 8-16 times; ie, those subtrees are of 3-4 levels at maximum

*If the voter voted at the pulling station, its Merkle tree should be still kept; whether as a zero value with certain flag or in a separate Verkle Tree this could be an implantation decision

-At the end the number of nodes is the Verkle Tree is the number of valid counted votes (n of a Verkle Tree is cryptographically proved), the counted votes are the last leaf node of each tree (could be zero if voted at pull station) and every deleted vote can be traced through its corresponding Merkle proof. Whatever the designer decision for handling ill-formed votes and those who voted at polling stations, the point is the old votes can be still traced and if the code is open source all the numbers can be cryptographically proved.

Summary & Conclusions

In this article we gave a political and technological historical brief on the development and status quo of the Estonian internet voting system. Then we explained the current system architecture and surveyed available material from the academic literature and different other available resources to cover reported attacks and/or vulnerabilities and how they were fixed. Last but not least, we discussed remaining risks and unfixed vulnerabilities; mainly not authenticating the voting application and not cryptographically proving the removal of "not to be counted votes" step. We suggested some possible solutions including the use of a Verkle Tree to aggregate votes where multiple votes from the same voter are to be accumulated in Merkle Tree whose root is a node in the Verkle commitment. It's worth mentioning that a similar approach maybe also feasible using STARKs as in [29].

References

[1]Piret Ehin, Mihkel Solvak, Jan Willemson, and Priit Vinkel, "*Internet voting in Estonia 2005–2019: Evidence from eleven elections*", Oct 2022; https://doi.org/10.1016/j.giq.2022.101718; https://www.sciencedirect.com/science/article/pii/S0740624X2200051X

[2] https://osce.org/files/f/documents/f/f/551179_0.pdf

[3]https://gafgaf.infoaed.ee/en/posts/great-divide-in-evoting/; last accessed 14/3/2024.

[4]https://ausadvalimised.ee/docs/yhisavaldus2023/ ; and their GitHub link, https://github.com/vaatlejad/vaatlejad.github.io

[5] "*A computer scientist made available the code for e-elections, which the electoral service has so far been fiercely hidin*g", https://digi.geenius.ee/eksklusiiv/arvutiteadlane-tegi-kattesaadavaks-e-valimiste-koodi-mida-valimisteenistus-on-seni-kiivalt-varjanud/; last accessed 2/1/2024.

[6] https://gafgaf.infoaed.ee/en/posts/perils-of-electronic-voting/; last accessed 4/1/2024.

[7]https://media.ccc.de/v/37c3-12298-should_e-voting_experience_of_estonia_be_copied#t=965; last accessed 15/1/2024.

[8] “The use of e-voting should be limited, https://arvamus.postimees.ee/7974894/mart-poder-e-haaletuse-kasutust-tuleks-piirata; last accessed 13/3/2024.

[9] Stone Bridge, "*About the destruction of Eestlus on the example of the Central Party*", https://uueduudised.ee/arvamus/kivisildnik-eestluse-havingust-keskerakonna-naitel/?fbclid=IwAR21qsnu8mI8GoBqgL-FLeQsiC46LBa6XZro71W0XDE8p-AJzM8EP_O24kE; last accessed 14/1/2024.

[10] https://ec.europa.eu/digital-building-blocks/wikis/pages/viewpage.action?pageId=533365949; last accessed 28/12/2023, https://scoop4c.eu/cases/estonian-internet-voting; last accessed 22/11/2023.

[11] https://www.valimised.ee/en/archive/statistics-about-internet-voting-estonia; last accessed 29/2/2024.

[12] Tartu University, https://skytte.ut.ee/en/content/register-mooc-internet-voting-13-october-2023; last accessed 22/3/2024.

[13] Arnis Parsovs, "*Estonian Electronic Identity Card: Security Flaws in Key Management*"; video https://www.usenix.org/conference/usenixsecurity20/presentation/parsovs

[14] Arnis Parsovs, "*Estonian Electronic Identity Card: Security Flaws in Key Management*", 29th USENIX Security Symposium, Aug 2020, 978-1-939133-17-5.

[15] Arnis Parsovs, "*Estonian Electronic Identity Card and its Security Challenges*", PhD Thesis, University of Tartu.

[16] Geenius. The police discovered 15,000 faulty ID cards, over 300 have been used (in Estonian), June 2019. https://digi.geenius.ee/rubriik/uudis/politsei-avastas-15-000-veaga-id-kaartiule-300-on-kasutatud/.

[17] https://e-estonia.com/estonia-introduced-a-new-id-card/; last accessed 20/3/2024.

[18] https://e-estonia.com/solutions/e-identity/id-card/; last accessed 20/3/2024.

[19] https://e-estonia.com/raulwalter-estonia-digital-identity-giant/; last accessed 20/3/2024

[20] https://valimised.ee/sites/default/files/2023-02/IVXV-protocols.pdf

[21]https://news.err.ee/1609194064/mobile-voting-likely-to-arrive-in-estonia-in-2025; last accessed 14/12/2023.

[22]https://www.researchgate.net/publication/373483642_Creating_a_Decryption_Proof_Verifier_for_the_Estonian_Internet_Voting_System

[23] Anggrio Sutopo, Thomas Haines, Peter Rønne. "*On the Auditability of the Estonian IVXV System and an Attack on Individual Verifiability*". Workshop on Advances in Secure Electronic Voting, May 2023, Bol, brac, Croatia. hal-04216242; https://halscience/hal-04216242

[24] https://github.com/valimised/ivotingverification/blob/published/app/src/main/java/ee/vvk/ivotingverification/util/ElGamalPub.java#L77-L83, and https://github.com/valimised/ios-ivotingverification/blob/published/VVK/Crypto.m#L141-L146; last accessed 20/2/2024.

[25] PMPhilip Daian, Tyler Kell, Ian Miers, and Ari Juels; July 2018; https://hackingdistributed.com/2018/07/02/on-chain-vote-buying/

[26] James Austgen, Andrés Fábrega, Sarah Allen, Kushal Babel, Mahimna Kelkar, Ari Juels, "*DAO Decentralization: Voting-Bloc Entropy, Bribery, and Dark DAOs*", Nov 2023; https://arxiv.org/abs/2311.03530 ; https://github.com/DAO-Decentralization/dark-dao/tree/main

[27]https://www.researchgate.net/publication/372570425_Individual_Verifiability_and_Revoting_in_the_Estonian_Internet_Voting_System

[28] https://github.com/DrShymaa2022/articles_papers/blob/main/Letter_to_Estonia_ivoting.pdf

[29]Max Harrison and Thomas Haines, "On the Applicability of STARKs to Counted-as-Collected Verification in Existing Homomorphically E-Voting Systems", Mar 2024; https://fc24.ifca.ai/voting/papers/Voting24_HH_On_the_Applicability_of_STARKs_to_Counted-as-Collected_Verification_in_Exisitng_Homomorphically_E-Voting_Systems.pdf