On practical aspects of coercion-resistant remote voting systems

Kristjan Krips[1,3] and Jan Willemson[1,2]

[1] Cybernetica AS, Ülikooli 2, 51003 Tartu, Estonia
{krisjan.krips,jan.willemson}@cyber.ee
[2] STACC, Ülikooli 2, 51003 Tartu, Estonia
[3] Institute of Computer Science, University of Tartu, J. Liivi 2, Tartu, Estonia

Abstract. Coercive behaviour is hard to control in the remote electronic voting setting. This is why a number of protocols have been proposed that aim at mitigating this threat. However, these proposals have remained largely academic. This paper takes the practical viewpoint and analyses the most common assumptions that are required by the various schemes, together with the exact level of coercion-resistance they provide.

1 Introduction

With introduction of Australian secret ballot into the voting process in mid-19th century, the threat of voter coercion was significantly reduced. Voting in a private booth surrounded by a controlled environment became the "gold standard" which has served democratic societies around the world well for over a 100 years.

However, several developments in recent decades have undermined the effect of Australian ballot as a coercion-resistance measure. First, technology of recording the private events within the voting booth (both with the voter cooperation and stealthily) has become readily available [3,17,18]. And second, human mobility has increased to an extent where expecting all the voters to come to a controlled environment on a particular day is less and less of an option [32].

These problems have motivated research and development in the field of coercion-resistant (remote) voting solutions. However, only a few of these solutions have actually been implemented in practice, leaving practical considerations such as usability or technical complexity of satisfying necessary assumptions often out of scope.

Another issue with the notion of coercion resistance is that it does not have a single clear interpretation. Thus it is not always immediately clear which levels of coercion resistance are achieved by various proposals.

The current paper aims at narrowing these gaps. We have selected seven different schemes from recent proposals and analyse them from two viewpoints. First, we identify common technical and organisational assumptions that these schemes rely on and assess their practical satisfiability. Second, we gather different interpretations of coercion resistance and analyse to what extent each one of the considered schemes achieves them.

We do not claim full coverage of all coercion-resistant schemes that have ever been proposed, but we have made an attempt to put together a representative selection of different approaches used for remote voting. Also, voting schemes often come in families. In this case we have selected members of such families for which coercion resistance and/or usability issues have been addressed the most.

2 Notions of voting freedom

One of the fundamental requirements of democratic elections is that the voter should be able to express her true preference freely, i.e. without being coerced. This broad statement has several possible interpretations, leading to more fine-grained requirements. E.g. following [4], we can identify the following properties.

- Basic *ballot privacy* guarantees that no one can learn how a voter voted (if she is not coerced and is willing to keep her vote secret). All the voting schemes studied in this paper satisfy this requirement.
- *Receipt-freeness* ensures that a malicious voter is unable to produce a proof for the value of her vote, making coercion essentially inefficient.
- *Coercion resistance* means intuitively that the voter should be able to cast a vote reflecting her true preference even if being monitored by the coercer for (most of) the voting period. To distinguish this property from the generic term, we will also call it *over-the-shoulder coercion resistance* in this paper.

Juels *et al.* [16] go even further and state three additional requirements that a fully coercion-free voting system should correspond to.

- The coercer should not be able to force the voter to abstain from elections.
- The coercer should not be able to force the voter to cast an invalid vote.
- The coercer should not be able to cast a valid vote if he gets access to the voter's credentials.

3 Coercion-resistant schemes and their assumptions

The threat of coercion depends on many aspects: type of elections, properties of the voting protocol, assumptions on the voting system and environment, awareness and coercibility of the voters, capabilities of the attacker, etc.

Typically, voting protocols aiming at some form of coercion resistance must make trade-offs between different goals. In the following, we describe and classify existing coercion resistant voting protocol proposals according to their assumptions, usability and applicability for different types of elections.

3.1 Re-voting based schemes / Estonian scheme

Re-voting is a metatechnique that can be used on top of other voting systems to provide voter with an option of changing her vote in case she was coerced

during the first attempts(s). An example of a pure re-voting-based protocol is the Estonian scheme, where this is the only anti-coercion measure in use [22].

The biggest problem with such schemes is that the coercer might stay with the voter until the end of the voting period (either physically or virtually [3]) to make sure that she does not cast a re-vote. To mitigate this threat (and also some other risks of remote voting), Estonia has chosen to end the Internet vote submission two hours before the polling stations are closed on the last day of advance voting period. The rationale is that if the voter feels coerced, she still has some time to submit her vote on paper and the paper vote cancels the e-vote. However, if the voter resides far from any of the polling stations (and enabling this scenario is one motivation of Internet voting), she can not submit an uncoerced vote. The whole system operates under the assumption that the share of such events is insignificant.

In addition, the re-voting functionality can affect integrity of the cast vote as an active attacker may use it to overwrite the previous vote.

On the positive side, enabling re-voting does not need extra setup on the client side, and the process is easy to understand for an average voter.

Aside from that, the Estonian system relies on significant technical assumptions, most notably voter credential pre-distribution. This is implemented via the national digital identity mechanisms (ID-card and mobile-ID), with the corresponding public keys being available via national PKI. Thus, even though the Estonian scheme relies on special client-side hardware, these devices are already very widely in use.

3.2 JCJ/Civitas family

Formal study of coercion resistance in voting systems was initiated in 2002 by Juels, Catalano and Jakobsson [15]. They gave a definition of coercion resistance and proposed the first scheme satisfying it, later becoming known as the JCJ scheme [16]. This research introduced fake credentials which the voter can use under coercion, but the coercer is unable to distinguish from the genuine ones.

In 2008, the JCJ scheme was extended by Clarkson, Chong and Myers by introducing distributed trust assumptions and improving the performance. The resulting protocol was called Civitas [8].

Neither of the JCJ and Civitas proposals specified how exactly the voter should select the appropriate credentials. Neumann and Volkamer noted in 2012 that this action is non-trivial, and may lead to both usability and security issues when implemented carelessly. Improving the specification of Civitas, they proposed an implementation based on smart cards and readers with PIN-pads and trusted displays [28]. Selection between a fake and a real credential would be accomplished by entering either a real or a fake PIN into the reader.

Essentially, Neumann-Volkamer proposal encapsulates all the critical voter-side operations into special hardware, which has to be trusted. While in principle such an approach can make credential handling more secure, it does not really move us much closer to a practical implementation. Smart card readers with

trusted preview are not commonplace on the market, and the smart cards would require a lot of non-standard functionality.

In a later research Neumann *et al.* have shown that, in principle, modern smart cards have sufficient performance required to implement such functions [27]. However, performance is not the only bottleneck in the practical deployment. The software implementing the protocol functionality needs to somehow get onto the cards.

Roughly speaking, election organisers have two approaches to tackle this problem. First, they can approach a large smart card vendor and convince it to implement the required functionality as part of the card firmware. Our interview with a representative of Gemalto (prevoius supplier of Estonian ID-cards) revealed that smart card vendors are quite reluctant to include limited-use applications on their products, and prefer implementing only general-purpose cryptographic primitives like standardised asymmetric signatures. One reason for this is that in many applications (likely including voting as well) the customers require certification, testing and validation of the security features of smart cards (for example, according to Common Criteria standard or FIPS-140-2). Such processes are expensive and time-consuming, and the vendor cannot earn this investment back selling limited-use cards.

Another option would be using programmable cards and implementing the functionality oneself in the spirit of [27]. The drawback of this approach is the need to support the whole software development life cycle locally. While it may give better control over the implementation, the risks are also higher. In case a bug is discovered, updating applications on the cards that have been distributed to numerous remote voters is a nightmare. Also, the whole expense of certification (in case it is desired) needs to be carried locally.

We conclude that while special-purpose smart cards provide an appealing option for a "poor man's HSM", their deployment has problems that are not necessarily easier to solve than the original challenge they were designed to meet.

As a part of the registration procedure, the Neumann-Volkamer protocol also depends on availability of anonymous channels (e.g. Tor is suggested by the authors). We refer to Section 3.4 for a more elaborate discussion on difficulties of achieving anonymous channels (using Tor) in practice.

Another branch of JCJ was developed by Araújo *et al.* in 2010 [2]. They introduced shorter credentials and provided a formal proof of coercion-resistance, although their proof relied on a non-standard number-theoretic assumption. In 2018, Neto *et al.* conducted usability studies for the CIVIS system [26], which is an implementation of the protocol proposed by Araújo *et al.* [2]. The study revealed that more than 90% of the test participants did not understand the functionality of casting fake votes. Also, they did not feel comfortable with the result, being unable to distinguish whether their submitted vote was real or fake. This brings the whole concept of using fake credentials under question.

3.3 Helios family

The original proposal of Helios by Adida [1] was explicitly targeted towards low-coercion environments. During later research, several extensions have been developed to enhance its coercion resistance.

KTV-Helios Kulyk, Teague and Volkamer have extended the Helios voting system to provide private eligibility verifiability, i.e. the property that anyone can verify that only votes from eligible voters are included in the tally, without revealing who actually submitted them [20,19]. As a by-product, they achieve receipt-freeness in the sense that the voter can not prove how she voted as she can undetectably re-vote. However, the authors stated that the protocol is susceptible to forced abstention and randomisation attacks. Following the authors' initials, the scheme is known as KTV-Helios.

The core idea of Kulyk *et al.* is to hide the true votes among dummy ones. Receipt-freeness is achieved allowing the voter to cast differential vote updates, so that the final vote would be a combination (e.g. product) of the votes cast. A similar approach was independently developed by Locher and Haenni [21].

Even though the dummy votes can be cast by any voter, most of them would probably not bother to do so. Hence a specific party called posting proxy or posting trustee is introduced by Kulyk *et al.*, and its task is to submit the dummy votes. In order to prevent timing side channels (see Section 3.4), posting trustee must operate in a randomised fashion.

Regarding the practical implementation aspects, the authors of KTV-Helios admit themselves that the understandability and usability issues remain largely unsolved [20]. Seeing many votes submitted onto the bulletin board on her behalf probably makes an average voter quite anxious. We add here a potential legal problem of voter impersonation, even if there are cryptographic proofs certifying that the extra votes do not change the final tally.

BeleniosRF In the original version of Helios, the voter can present encryption randomness as a receipt for the coercer. BeleniosRF uses re-randomisable ciphertexts and signatures, with part of the randomness being out of the voter's control, making it impossible for a voter to produce such a receipt [4].

The ballot is signed by the voter and re-randomisation of the ballot by the server does not invalidate the corresponding signature. Thus, the voter can verify the signature to make sure that the vote has not been changed. However, this applies only when re-voting is not enabled. The authors of BeleniosRF state that in case of re-voting the voters would not be able to check which of their ballots were re-randomised by the server. Therefore, BeleniosRF does not allow re-voting and thereby does not provide protection against over-the-shoulder coercion. However, vulnerability to in-person coercion is one of the major objections against remote electronic voting in the first place [10,11,25,24,13,14].

The authors of BeleniosRF argue that changing one's vote is a legally grey area anyway, and most of the countries would need to go through a complicated legal process before they can support it.

While we agree that legislative changes are necessary to support re-voting, we feel that the authors of BeleniosRF over-estimate the complexity of this process.

For example, extensive social and legal debate concerning constitutionality of re-voting took place in Estonia when Internet voting was introduced there. A few months before the first Internet-enabled elections, the President of Estonia brought Internet voting provisions to the Supreme Court for constitutional review, arguing that the possibility to change Internet votes gives advantages to Internet voters in comparison with paper voters. The decision of the Supreme Court did not support this point of view, reaching the conclusion that merely a technical option of casting multiple votes does not put Internet voters into any kind of advantage [23].

While the outcome of a similar legal discussion may be different in other jurisdictions, we feel that re-voting as an easy-to-implement and relatively efficient anti-coercion measure is important enough to review some of the legislative principles. Changing legislation in order to catch up with technological advancements is an unavoidable process anyway.

3.4 Selene

The primary design goal of the Selene scheme proposed by Ryan *et al.* [31] is achieving a user-friendly end-to-end vote verification protocol. As too strong of a verification mechanism brings along a threat of coercion, the authors of Selene have also paid a lot of attention to mitigating this threat. They propose using cryptographic tracking numbers which are first committed to a bulletin board using trapdoor commitments. After the end of the voting period, clear-text votes with clear-text tracking numbers are displayed on the bulletin board as well. The (voter-controlled) trapdoor can later be used to open the commitment to any tracking number of coercer's liking.

The voter, of course, still needs to somehow identify the real tracking number of her own vote. This is facilitated by sending her the correct decommitment value α. In order to fool the coercer, the voter can produce an alternative decommitment value α' that is cryptographically indistinguishable from α and points to any vote requested by the coercer.

However, cryptographic indistinguishability is not sufficient, as the attacker potentially has a number of side channels available to separate the true α from voter-generated α'-s. The authors of Selene acknowledge this problem and state that α-terms should be transferred over an unauthenticated and private channel.

Unfortunately, implementing such a channel is non-trivial. Note first that in order to mitigate the threat of coercion, it is not sufficient just to drop strong authentication mechanisms like signatures. For example, if α (or its shares coming from the trustees) is sent via regular, otherwise unauthenticated email, it has to carry sender's email address. There are both legal and usability issues that suggest using a fixed official address rather some randomly generated ones. Email is just an example here, similar problems would occur if other taggable delivery channels like instant messaging or web bulletin board would be used.

In principle, the process of preparing a false α' can also include sending it from the official address. In this case there is still the timing side channel that the coercer can use to distinguish the genuine α. In order to counter this, the

genuine α-s would need to be sent at randomised moments, and the voter must prepare α' during this period. This is doable and is also proposed by the authors, but it complicates the voter's view of the protocol substantially.

We can also imagine genuine α-s being sent out via regular mail, printed on standard office paper. The voter can print α' out on her home printer, but this assumes using exactly the same kind of paper, printing resolution, etc. In addition, majority of modern colour laser printers mark the printed papers with tracking dots which can be used to identify the printer [30]. We can see that it could be possible to deliver α-s with the help of the postal service, but generating the fake values is not as easy as the authors of Selene probably foresaw. Note also that a vote buyer is typically after a number of votes and he can live with some of the voters being able to fool him as long as their share is not too high.

One can also utilise stronger anonymisation techniques, e.g. mixing or onion routing. These would only help if the full set of messages is larger than just the official α-terms, as otherwise we would have no sender anonymity. One may consider using an existing anonymisation network, say, Tor (as also recommended by Neumann and Volkamer [28]). However, due to significant illegal activity happening over it, utilising Tor for legally binding elections would be questionable.

We argue that this dilemma is at least partially inherent and not specific to Tor. On one hand, too small of an anonymisation set does not fulfil the goal, but fighting doubtful traffic in a large network is practically impossible.

Furthermore, by relying on Tor (let's still use it as a prime example) new problems are introduced. Referring to the objectionable content and general uncontrollable nature, several countries have attempted to block/filter Tor traffic[4]. This makes it hard for expatriates living in those countries to participate in the elections remotely, but supporting expatriate participation is one of the main reasons for introducing remote electronic voting in the first place.

Setting up private channels from the election organiser to all the voters is not a trivial task either. As Selene already relies on a PKI for vote signing, assuming additional access to an authentic public-private key pair for encryption and decryption is probably not a big extra. However, even the α-term encrypted with the voter's public key has to be delivered to her somehow. We conclude that channel privacy does not really help against the soft sender identification problem described above.

In 2019, Distler *et al.* performed an e-voting usability study based on a Selene protocol implementation [9]. Unfortunately, they left the steps related to coercion resistance (including preparing the fake α' and selecting it in the presence of the coercer) out of scope. We also note that their implementation relies only on a mobile device for both vote casting and verification. This means that verification is inefficient against the malicious device and does not thus fulfil the purpose of verification. We feel that in order to get a more realistic understand-

[4] It is hard to get reliable statistics on the extent of Tor filtering, but there exists indirect evidence in the form of the share of users relying on Tor bridges (`https://metrics.torproject.org/userstats-bridge-table.html`) and observed irregularities (`https://metrics.torproject.org/userstats-censorship-events.html`).

ing of usability of Selene protocol the authors of [9] should have implemented a complete version, e.g. by using a second channel for verification. Adding extra channels and steps would have likely changed the user perception and feedback.

3.5 Eos

Patachi and Schürmann have proposed the Eos voting scheme based on a specific flavour of ring signatures, namely conditional linkable ring signatures [29]. As each voter can have multiple pseudo-identities in the scheme, conditional linkability allows the signer to choose if the signatures can be linked to the same identity by the verifier.

There are two main anti-coercion measures in Eos. First, the voter can use subliminal hinting (called selecting between "red" and "green" envelopes or alternative pseudo-identities in [29]) while preparing the encrypted vote. In practice, such hinting would be implemented by presenting either a real or pseudo-PIN to a special-hardware voting device or to the coercer who controls the device.

Second, if the actively coerced voter had to cast a vote using a valid PIN, she may later re-vote to update the vote. However, in that case the public bulletin board will contain multiple encrypted votes given by the same pseudo-identity, which may be known to the coercer. In that case, the voter may have to lie to the coercer that the coercer was the last one to cast the vote.

The protocol makes several non-trivial assumptions. First, to get rid of side-channels during submitting the ring-signed votes, one would need to use anonymous channels, but achieving these is quite tricky in practice (see Section 3.4).

Second, special hardware tokens would be needed to implement the client-side operations (key management, PIN validation, identity selection, and signature computation). The paper [29] suggests that hardware wallets designed for storing the keys for cryptocurrencies could be used in this role. It might be possible to reprogram such hardware, but distributing the hardware or the private keys to the voters is a non-trivial task.

As the selection between identities would happen by entering a real or pseudo-PIN, we also have all the regular problems of pseudo-PIN management – if the user enters a wrong PIN, the device can not give any feedback (as the coercer might be watching), and would quietly submit a vote that the voter did not intend to (e.g. in the scenario where the voter wanted to use a pseudo-PIN, but accidentally used a real one).

3.6 Selections

A special form of fake credentials called *panic passwords* has been proposed by Clark and Hengartner in 2008 [5]. The essence of panic passwords is what the name says – the user can select a true password together with a set of alternative ones that can be used to covertly alert the system that the user is under abnormal circumstances, e.g. coercion.

The latter is an important threat scenario in case of remote voting, so the same authors have built a coercion-resistant voting scheme called Selections around their core idea [6].

Unfortunately, making human-memorisable passwords to work as fake credentials is even more problematic than in case of cryptographic credentials.

First, a complex registration process is needed. Of course, it has to take place in a controlled, coercion-free environment, but this is a standard assumption. The registration procedure can even be implemented bare-handed (i.e. not requiring the voter to perform computations by heart). An Internet-enabled computer is still required inside the controlled registration booth to print out a voter preparation sheet. This is meant as a countermeasure "... in the event that an adversary ensured she entered the registration process without her sheet" [6].

The only way the coercer can achieve this is to search through the voter's belongings and walk together with her until the door of the registration booth. But if the coercer is prepared to do this much, he can also request the voter to record all her actions with a camera or even send a live stream [3]. As a result, the effect of controlled registration environment will be significantly reduced.

During the registration process, the previously selected and encrypted panic passwords are re-randomised. The voter selects one of the re-randomised encryptions which is posted to a public roster. It is assumed in the protocol that the voter deletes the randomness used for re-randomisation and does not record it. Building security properties on the assumption that some value is deleted is always questionable. There may exist side channels that the coercer forces the voter to use to record or stream the value. If the coercer took part in creating the voter preparation sheet and has access to it, then the re-encrypted panic password on the public roster can be matched with the encrypted panic password on the preparation sheet. Thus, the randomness gives a way to prove the validity of the password given to the coercer.

We also noticed that the registration protocol differs significantly when comparing the full paper (e-print) [7] to the conference paper [6]. In the e-print version, the registration protocol allows the voter to rewind the process back to the re-randomisation phase. In the conference paper, the registration protocol allows the voter to rewind the process back to the beginning, i.e., to selecting new panic passwords. However, the difference is important as some of the coercion protections depend on the rewinding functionality.

Additionally, Selections suffers from the typical problems of password-based systems. Even though [6] proposes measures to increase password memorability, the scenario of voting stretches these boundaries. The idea of [6] was to go through the complex registration process once and then use the credentials over several events. However, elections typically only happen once in a few years, and many voters are likely to forget their passwords over this time, no matter how good of a mnemonic is used. To counter this problem, humans tend to write the passwords down, increasing their coercibility as a result.

4 Other coercion properties

In this section, we discuss the extra coercion properties (i.e. forced abstention, casting an invalid vote, and forced surrender of credentials) of the schemes.

A voter can be forced to not take part in the elections if a coercer has a way to check if the voter abstained from voting. As potential attackers, we also consider corrupt election officials and democratically elected politicians who decide to deviate from fair election practices. Such an attacker would be able to to indirectly manipulate a large portion of the electorate.

Forcing the voter to cast an invalid vote can benefit the coercer in (at least) two ways. First, in case the voter is supporting a party opposing the coercer's views, the invalid vote would have no effect and the voter would effectively abstain from the elections. Second, if the invalid vote would be posted to a bulletin board, the attacker could remotely check if the voter behaved according to the instructions. Even if the invalid vote would not be published, it may still be possible that election officials are able to see the value of the vote and thus be able to play the role of a coercer.

In case another person would be able to use the voter's credentials, it would be possible to cast the vote on behalf of the voter. Juels *et al.* [16] refer to this type of an attack as a *simulation attack*.

The rest of this Section is devoted to the discussion of these coercion properties. Table 1 summarises the main assumptions used by different coercion-resistant protocols proposals together with their level of coercion-resistance in respect to the requirements listed in Section 2. The only exception is the basic ballot privacy that all the considered schemes trivially satisfy.

4.1 Re-voting based schemes / Estonian scheme

The Estonian voting system provides protection against standard versions of these coercion attacks. More specifically, an outside third party is not able to detect if a voter cast a vote online or abstained as there is no public proof of the vote casting. There is a private bulletin board in the Estonian voting system, which is only accessible to the election officials and auditors. The official voting client software does not support casting an invalid vote. Finally, the signing key of the voter is stored inside of a smart card, hence the coercer would need to have physical access to use the credentials.

However, the situation gets more complicated in case of an attacker who has insider information. The voting system has to verify the ballot signatures to make sure that only the votes of eligible voters are accepted. Thus, insiders could check if a certain voter abstained.

There is also an insider threat when an invalid vote is cast. To cast an invalid vote, either the voter or the coercer would have to create a non-standard voting client. In case the invalid vote would have a correct format and would correspond to a non-existing candidate number of a suitable district, the vote would be decrypted during the tallying process. Writing a voting client that would allow casting such votes is possible as the voting protocol and the communication API

Table 1. Cross-table of assumptions and achieved coercion resistance properties

	Estonia	NV-Civitas	KTV-Helios	BeleniosRF	Selene	Eos	Selections
Special client hardware	●[1]	●	●	○	○	●	○
Anonymous channels	○	●	●	○	●	●	●
PKI / key distribution	●	●[2]	●	●	●[2]	●[2]	○
Subliminal password/PIN hinting	○	●	○	○	○	●	●
Casting a re-vote	●	●	●	○	◑[3]	●	●
Non-trivial registration	○	◑[4]	○	○	○	○	●
Receipt-freeness	○	●	●	●	◐[5]	●	◐[6]
Over-the-shoulder coercion resistance	●	●	◐[7]	○	◐[8]	●	●
Resistance to forced abstention	◐[9]	●	◐[10]	○	◑[11]	●	◐[12]
Resistance to casting an invalid vote	◐[9]	●	◐[13]	◑[14]	◑[15]	◑[16]	◑[17]
Resistance to simulation attack	◐[18]	●	◐[19]	○	○[20]	◑[21]	●[22]

● = is assumed / holds ○ = is not assumed / does not hold ◐ = may hold
◑ = depends on the implementation

[1] Smart card based ID-cards are mandatory in Estonia and widely in use.
[2] PKI is not explicitly mentioned, but its functionality is implicitly described.
[3] Whether re-voting is allowed in Selene depends on the used policy [31].
[4] Information about the registration process of NV-Civitas can be found in [28].
[5] Selene's receipt-freeness depends on the anonymous channel, see Section 3.4.
[6] Whether Selections is receipt free depends on how the re-randomisation randomness is handled during registration. For more information, see Section 3.6.
[7] The property depends on how the coercer prevents re-voting, see Section 4.3.
[8] The property depends on the re-voting policy in the implementation of Selene [31].
[9] The attack can be implemented by an insider, see Section 4.1.
[10] KTV-Helios is susceptible to forced abstention only in the case of an active attacker.
[11] For information about the implementation of Selene, see Section 4.4.
[12] It is not clear whether Selections is resistant to forced abstention, see Section 4.6.
[13] In KTV-Helios invalid votes can be cast, but they will be removed by plaintext equality tests before votes are published in the bulletin board.
[14] See Section 4.3 for information about the coercion properties of BeleniosRF.
[15] Vote casting procedure is not specified in Selene, see Section 4.4 for more details.
[16] Whether it is possible to cast an invalid vote depends on the version of Eos. More information can be found from Section 4.5.
[17] It is not specified how vote is encoded and how votes are tallied in Selections [6].
[18] The coercer might be able to get physical access to the smart card. However, it is possible to re-vote as described in Section 4.1.
[19] The coercer might be able to get physical access to the smart card. However, the voter may be able to re-vote to cancel the coerced vote as described in Section 4.3.
[20] It is not specified how keys are managed in Selene [31]. In case Selene is used as an add-on, then key management may be specified by the underlying voting protocol.
[21] The possibility of casting a valid vote with the voter's HSM depends on the configuration of the HSM. For more information, see Section 4.5.
[22] If the registration process and thus the credentials are remotely monitored then the voter has the option to revoke the registration and vote in person. For more information, see Section 4.6.

is public. Now, if a voter would be able to cast such an invalid vote then either the members of the election committee or the auditor who audits the election result might be able to read the invalid value. Thus, the coercer would have to cooperate with the election officials or the auditor to see the vote value.

In order to get a hold of the signing keys, the coercer would have to take the possession of all of the digital ID-s of the voter along with the corresponding PIN codes. Still, the voter could use a non-digital ID to cast a paper vote in the polling station that overwrites the e-vote. Thus, all non-digital ID-s would also have to be collected by a coercer in case the coercer would like the voter to abstain from participating in the elections. Such an attack could be applied on selected individuals, but this approach does scale.

4.2 NV-Civitas

NV-Civitas was the only one of the protocols that we analysed not susceptible to the three aforementioned coercion attacks. Forced abstention is impossible as the ballots are not signed and are delivered over an anonymous channel. Invalid ballots are either rejected by the smart card or by the voting system after checking the proof of vote well-formedness [28]. It is also impossible to force the voter to surrender the credentials as the voter can give the coercer the smart card with a fake PIN, which would create a ballot with invalid credentials.

4.3 Helios family

KTV-Helios While it is possible to cast invalid ballots in KTV-Helios, they do not end up on the bulletin board. Invalid ballots are removed before tallying with the help of plaintext equality tests. Thus, invalid votes are not decrypted.

The authors state that casting an invalid vote can cause the voter to abstain from the elections. The attack would always work in an active scenario where the attacker waits until the end of the voting period to force the voter to cast an invalid vote. In this case the invalid vote would be discarded before the tally and the voter would not have enough time to re-vote. However, if the value of the invalid vote would be known to the voter and there would be time to re-vote, the voter may be able to cancel the previous vote.

As the signing keys are stored on smart cards, it is in principle possible to force the voters to give up the cards, but such an attack would not scale well.

BeleniosRF BeleniosRF uses a fixed message space for encoding the vote and tallying is done homomorphically. Thus, the possibility of casting an invalid vote depends on the implementation. In case the message space is not used up to encode the candidates, it might be possible to cast an invalid vote that would be published.

The other two coercion attacks could be applied in the case of BeleniosRF. It is possible to force the voter to abstain from voting as there is public proof of participation in the voting event. The signature of the randomised public ballot can be verified by the voter. In case the voter's public key is accessible to the coercer, the latter is able to verify all the ballots on the bulletin board. Also, the

voter ID is verified before a ballot is accepted and re-randomised by the bulletin board. Thus, the election officials could coerce voters to abstain.

A coercer might also be able to force the voter to surrender her secret key as no special hardware is used for storing the secret key. However, the voter is only able to give one vote, so the coercer would have to get access to the signing key before the voter casts her vote.

4.4 Selene

Whether Selene is safe from forced abstention attack depends on the implementation of the protocol. The basic scheme is vulnerable as the ballots signed by the voters are published on the bulletin board. However, the optional enhancement of using pseudonymous credentials enables giving signatures without revealing the identity of the voter. Thus, the extended scheme is resistant to forced abstention attack if the coercer can not access the voter's pseudonymous credentials.

Similarly, the ability to cast an invalid vote depends on the implementation of the vote casting procedure and is not fixed on the protocol level. Selene can be used as an add-on on top of another voting system, which may remove invalid votes. E.g., Selene combined with JCJ is resistant to casting an invalid vote [12].

Still, Selene is susceptible to forced surrender of credentials as no hardware token is proposed for storing the secret key. Also, re-voting policy is not fully specified, thus it is not clear if voter's initial choice could be overwritten.

4.5 Eos

Eos is resistant to the forced abstention attack. It uses ring signatures to hide voter identities from the election officials. Also, an anonymous channel is used to cast the vote. Thus, it won't be possible to detect if a specific voter has voted.

The authors of Eos acknowledge that in the basic version of the protocol a coercer could force a voter to cast an invalid vote [29]. As a solution, they propose using a disjunctive zero-knowledge proof protocol, such that the voter could prove that her vote is in the set of valid votes. In that case, invalid votes could be removed before they are tallied and published.

It would be difficult to force a voter to surrender the credentials as that would require getting physical access to the voter HSM. However, the possibility can not be excluded as it is not clear if the correct PIN code could be extracted from the voter or HSM. It might be possible to try out all PIN code combinations in order to give a valid vote. It is also not specified in [29] if the HSM would allow to change the valid PIN codes. A successful change of the PIN would probably reveal the real PIN code. If changing PIN codes is not possible, then the usability aspect of the HSM would come under question. Even if the coercer could use the HSM, the attack would not scale well.

4.6 Selections

It is not clear whether Selections is resistant to the forced abstention attack. While the votes are cast over an anonymous channel and the passwords are re-randomised, there are some questions that can not be answered based on the

protocol description. First, the protocol allows to revoke voter registration before pre-tallying, but it is not specified how it could be implemented. The authors of Selections also state that the revocation process might not be covered by coercion resistance. Second, during the registration, the randomised encryption of the password is posted to the roster along with the VoterID. However, it is not stated what the VoterID is or how it is assigned to the voters. Thus, the coercer might be able to use the VoterID to check if the coerced voter registered to use Selections. Third, it is assumed that during the registration process, the voter does not copy or remember the randomisation of the selected password. However, modern technology makes it quite easy to copy and broadcast information. Rewinding some of the registration steps would not help in case the coercer forces the voter to live broadcast the process.

The protocol does not specify the way how the vote is represented or how the votes are tallied. Thus, the possibility of casting an invalid vote depends on the specific implementation of the protocol.

If the coercer would like to get access to the valid credentials, the voter would have to record or broadcast the registration process. However, in that case the voter could revoke the registration before pre-tallying and thus invalidate the credentials given to the coercer together with the vote. After revoking, the voter could go to the polling station to vote in person.

5 Conclusions and further work

Developing a voting protocol to meet the requirements of a given jurisdiction is a complex task. On one hand, we would like the protocol to be secure against all critical attacks, but this security comes with a price of increased implementation complexity and technical assumptions that need to be satisfied.

This paper focused on coercion-resistance properties of various voting protocols proposed in academic literature from the practical system developer viewpoint. As academic proposals are not required to include real-life deployments, it is very easy to leave some of the implementation details out of consideration. Unfortunately, there are many devils hidden in these details.

During our research we identified six main (groups of) popular technical assumptions. Some of them (like existence of PKI or ability to cast a re-vote) indeed have readily accessible practical instantiations. At the same time, the requirements to set up anonymous channels or distribute special-purpose client hardware are easy to write down on paper, but quite tricky to implement.

Subliminal hinting using fake credentials is one of the oldest methods to achieve provable coercion-resistance properties, but a recent usability study by Neto *et al.* [26] found that more than 90% of the test participants did not understand this functionality. This questions the whole idea of using fake credentials.

In general, there is a lack of usability studies that focus on the coercion-resistance aspects of voting protocols. We see this as an important open question that requires further research.

Another general shortcoming of the current proposals is under-specification. On several occasions, it was impossible to determine susceptibility to certain

attacks as this would have depended on specific implementation aspects. Sure, a 16-page academic paper can not fit all the details, but we encourage future scholars to accompany their proposals with deployed implementations. This would help identifying potential problems in an earlier stage of academic discussion.

Acknowledgments The research leading to these results has received funding from the Estonian Research Council under Institutional Research Grant IUT27-1 and the European Regional Development Fund through the Estonian Centre of Excellence in ICT Research (EXCITE) and the grant number EU48684.

References

1. Adida, B.: Helios: Web-based Open-Audit Voting. In: Proceedings of the 17th USENIX Security Symposium. pp. 335–348. USENIX Association (2008)
2. Araújo, R., Rajeb, N.B., Robbana, R., Traoré, J., Yousfi, S.: Towards practical and secure coercion-resistant electronic elections. In: CANS 2010, Proceedings. LNCS, vol. 6467, pp. 278–297. Springer (2010)
3. Benaloh, J.: Rethinking voter coercion: The realities imposed by technology. USENIX Journal of Election Technology and Systems (JETS) 1, 82–87 (2013)
4. Chaidos, P., Cortier, V., Fuchsbauer, G., Galindo, D.: BeleniosRF: A Non-interactive Receipt-Free Electronic Voting Scheme. In: Proceedings of 2016 ACM CCS. pp. 1614–1625. ACM, New York, NY, USA (2016)
5. Clark, J., Hengartner, U.: Panic Passwords: Authenticating under Duress. In: HotSec'08, Proceedings. USENIX Association (2008), `http://www.usenix.org/events/hotsec08/tech/full_papers/clark/clark.pdf`
6. Clark, J., Hengartner, U.: Selections: Internet Voting with Over-the-Shoulder Coercion-Resistance. In: Danezis, G. (ed.) FC 2011, Revised Selected Papers. LNCS, vol. 7035, pp. 47–61. Springer (2011)
7. Clark, J., Hengartner, U.: Selections: Internet voting with over-the-shoulder coercion-resistance. Cryptology ePrint Archive, Report 2011/166 (2011), `https://eprint.iacr.org/2011/166`
8. Clarkson, M.R., Chong, S., Myers, A.C.: Civitas: Toward a Secure Voting System. In: 2008 IEEE Symposium on Security and Privacy (S&P 2008). pp. 354–368. IEEE Computer Society (2008)
9. Distler, V., Zollinger, M.L., Lallemand, C., Roenne, P.B., Ryan, P.Y.A., Koenig, V.: Security – Visible, Yet Unseen? In: CHI 2019. pp. 605:1–605:13. ACM, New York, NY, USA (2019)
10. Gerck, E., Neff, C.A., Rivest, R.L., Rubin, A.D., Yung, M.: The Business of Electronic Voting. In: FC 2001, Proceedings. LNCS, vol. 2339, pp. 234–259. Springer (2001)
11. Hoffman, L.J., Cranor, L.F.: Internet voting for public officials: introduction. Commun. ACM 44(1), 69–71 (2001)
12. Iovino, V., Rial, A., Rønne, P.B., Ryan, P.Y.A.: Using Selene to Verify Your Vote in JCJ. In: Financial Cryptography and Data Security. LNCS, vol. 10323, pp. 385–403. Springer (2017)
13. Jefferson, D.R., Rubin, A.D., Simons, B., Wagner, D.A.: Analyzing internet voting security. Commun. ACM 47(10), 59–64 (2004)
14. Joaquim, R., Ribeiro, C., Ferreira, P.: Improving Remote Voting Security with CodeVoting. In: Towards Trustworthy Elections, New Directions in Electronic Voting. LNCS, vol. 6000, pp. 310–329. Springer (2010)

15. Juels, A., Catalano, D., Jakobsson, M.: Coercion-Resistant Electronic Elections. Cryptology ePrint Archive, Report 2002/165 (2002), https://eprint.iacr.org/2002/165
16. Juels, A., Catalano, D., Jakobsson, M.: Coercion-resistant electronic elections. In: Proceedings of WPES 2005. pp. 61–70. ACM (2005)
17. Krips, K., Willemson, J., Värv, S.: Implementing an Audio Side Channel for Paper Voting. In: E-Vote-ID 2018, Proceedings. LNCS, vol. 11143, pp. 132–145. Springer (2018)
18. Krips, K., Willemson, J., Värv, S.: Is your vote overheard? A new scalable side-channel attack against paper voting. In: Proceedings of Euro S&P 2019. pp. 621–634. IEEE (2019)
19. Kulyk, O.: Extending the Helios Internet Voting Scheme Towards New Election Settings. Ph.D. thesis, Technische Universität Darmstadt (2017)
20. Kulyk, O., Teague, V., Volkamer, M.: Extending Helios Towards Private Eligibility Verifiability. In: VoteID 2015, Proceedings. LNCS, vol. 9269, pp. 57–73. Springer (2015)
21. Locher, P., Haenni, R.: Receipt-free remote electronic elections with everlasting privacy. Annals of Telecommunications 71(7) (Aug 2016)
22. Madise, Ü., Martens, T.: E-voting in Estonia 2005. The first Practice of Country-wide binding Internet Voting in the World. In: Krimmer, R. (ed.) Electronic Voting 2006. LNI, vol. 86, pp. 15–26. GI (2006)
23. Madise, Ü., Vinkel, P.: Internet voting in Estonia: from constitutional debate to evaluation of experience over six elections. In: Regulating eTechnologies in the European Union. Normative Realities and Trends, pp. 53–72. Springer (2014)
24. Mitrou, L., Gritzalis, D., Katsikas, S.K.: Revisiting Legal and Regulatory Requirements for Secure E-Voting. In: SEC2002. IFIP Conference Proceedings, vol. 214, pp. 469–480. Kluwer (2002)
25. Mohen, J., Glidden, J.: The case for internet voting. Commun. ACM 44(1), 72–85 (2001)
26. Neto, A.S., Leite, M., Araújo, R., Mota, M.P., Neto, N.C.S., Traoré, J.: Usability Considerations For Coercion-Resistant Election Systems. In: Proceedings of the 17th Brazilian Symposium on Human Factors in Computing Systems. pp. 40:1–40:10. IHC 2018 (2018)
27. Neumann, S., Feier, C., Volkamer, M., Koenig, R.: Towards A Practical JCJ / Civitas Implementation. Cryptology ePrint Archive, Report 2013/464 (2013), https://eprint.iacr.org/2013/464
28. Neumann, S., Volkamer, M.: Civitas and the Real World: Problems and Solutions from a Practical Point of View. In: ARES 2012. pp. 180–185. IEEE (2012)
29. Patachi, S., Schürmann, C.: Eos a universal verifiable and coercion resistant voting protocol. In: E-Vote-ID 2017, Proceedings. LNCS, vol. 10615, pp. 210–227. Springer (2017)
30. Richter, T., Escher, S., Schönfeld, D., Strufe, T.: Forensic analysis and anonymisation of printed documents. In: Proceedings of IH&MMSec '18. pp. 127–138. ACM, New York, NY, USA (2018)
31. Ryan, P.Y.A., Rønne, P.B., Iovino, V.: Selene: Voting with Transparent Verifiability and Coercion-Mitigation. In: FC 2016 International Workshops, Revised Selected Papers. LNCS, vol. 9604, pp. 176–192. Springer (2016)
32. Willemson, J.: Bits or paper: Which should get to carry your vote? Journal of Information Security and Applications 38, 124–131 (2018)