MASTER THESIS INFORMATION SCIENCE

Author: M.W. Verwoert (Marc-William Verwoert)
Student number: S4718801

Date: 24/08/2021

Radboud University
Nijmegen, the Netherlands
The Faculty of Science

Cyberattack scenarios for remote electronic voting consultations using IRMA

First Supervisor:

dr. B.E. van Gastel (Bernard)
Faculty of Science, Radboud University Nijmegen, the Netherlands

Second Supervisor:

dr. ir. E. Poll (Erik)
Faculty of Science, Radboud University Nijmegen, the Netherlands

Samenvatting

Via raadplegingen kunnen burgers hun mening en wensen kenbaar maken over bepaalde besluiten van gemeenten. Nederlandse gemeenten kampen met een lage opkomst bij raadplegingen gehouden op papier. Door de lage opkomst bij raadplegingen zijn de resultaten vaak niet representatief. Om de opkomst te verhogen kan elektronisch stemmen op afstand worden gebruikt in Nederlandse gemeente raadplegingen als alternatief voor stemmen op papier.

Een nieuw elektronisch afstand stem platform in Nederland, genaamd **IRMA vote**, presenteert zichzelf als een veilige en praktische optie voor kleinschalige raadplegingen. De kern van dit elektronisch afstand stem platform is het splitsen van het stemproces over twee afzonderlijk gehoste servers om het stemgeheim te garanderen, het zogenaamde **split server-ontwerp**. De eerste server controleert de stemgerechtigdheid (en kent dus de identiteit van de stemmer), terwijl de tweede server alleen de uitgebrachte stemmen registreert (en dus de identiteit van de stemmer **niet** kent).

Toch is er op dit moment geen onderzoek bekend dat cyberaanval scenario's schetst bij kleinschalige raadplegingen die gebruik maken van elektronische stemmen op afstand. Daarnaast is er beperkt onderzoek gedaan naar de vraag of elektronische afstand stem platformen met het gesplitste server ontwerp robuust genoeg zijn om cyberaanvallen in kleinschalige raadplegingen te weerstaan. Deze twee punten komen in dit onderzoek aan de orde.

Om deze redenen identificeert dit onderzoek relevante dreigingen, aanvaller modellen en cyberaanval scenario's voor raadplegingen en stelt beveiligingseisen voor dergelijke raadplegingen vast, voornamelijk op basis van het **stemgeheim** en het **voorkomen van dubbel stemmen**. Deze beveiligingseisen worden gebruikt om de impact van elk cyberaanval scenario te bepalen. Op basis van de cyberaanval scenario's worden risico's (de waarschijnlijkheid van een cyberaanval scenario keer de impact van dit scenario) bepaald voor elektronische afstand stem platformen met een gesplitst server ontwerp gebruikt in kleinschalige raadplegingen. Op deze manier probeert dit onderzoek de cyberveiligheid van elektronisch stemmen op afstand in raadplegingen te vergroten.

Aan de hand van de resultaten van dit onderzoek raden we aan dat elektronische stem platformen niet worden gebruikt bij politieke verkiezingen, aangezien elektronische stem platformen niet kunnen voldoen aan stemvrijheid en we beschouwen deze vereiste als een cruciale factor voor politieke verkiezingen. Echter, hoe meer een raadpleging kan worden geclassificeerd als '**kleinschalig**' of met '**kleine belangen**,' hoe meer het risico van elk cyberaanval scenario wordt beperkt door het gesplitste server ontwerp van het elektronische stem platform op afstand.

Ter conclusie, we hebben in dit onderzoek cyberaanval scenarios, hun risico's en relevante beveiligingsvereisten geclassificeerd voor raadplegingen. Onze resultaten hebben aangetoond dat **elektronische afstand stem platformen met een gesplitst server ontwerp** met beheersbare risico’s kunnen worden gebruikt in kleinschalige raadplegingen met kleine belangen.

Abstract

Consultations allow citizens to voice their opinions and wishes on certain decisions made by municipalities. Dutch municipalities suffer from low voter turnout in traditional paper-based consultations. Low voter turnout in consultations causes the results to be often unrepresentative. Remote electronic voting can be used as an alternative to paper-based voting in Dutch municipality consultations to increase voter turnout.

A new remote electronic voting platform in the Netherlands, **IRMA vote**, presents itself as a safe and practical option for small-scale consultations. The core of this remote electronic voting platform is splitting the voting process over two separately hosted servers to guarantee voter secrecy, termed the **split server design**. The first server checks eligibility (and therefore does know the voter's identity), while the second server only registers the cast votes (and therefore does **not** know the voter's identity).

Nevertheless, there is currently no known research that outlines attack scenarios in small-scale remote electronic voting consultations. Additionally, there is limited research on whether remote electronic voting platforms with the split server design are robust enough to withstand cyberattacks in small-scale consultations. These two points are addressed in this research.

Hence, this research identifies relevant threats, attacker models, and cyberattack scenarios for consultations and establishes security requirements for such consultations mainly based on the **secrecy** requirement and **double voting prevention**. These security requirements are used to determine the impact of each cyberattack scenario. Based on the cyberattack scenarios, risks (the likelihood of the cyber attack scenario times its impact) are determined for remote electronic voting platforms with a split server design used in small-scale consultations. In this manner, this research aims to increase the cybersecurity of remote electronic voting in consultations.

Based on the results of this research, we recommend that electronic voting platforms should not be used in political elections since electronic voting platforms can not meet the liberty requirement, and we consider this requirement a crucial factor for political elections. However, the more a consultation can be classified as '**low-stakes**' or '**small-scale**,' the more the risk of each cyberattack scenario is limited by the split server design of the remote electronic voting platform.

In conclusion, we have classified cyberattack scenarios, their risks, and relevant security requirements in consultations in this research. Our results have shown that **remote electronic voting platforms with a split server design** could be used in small-scale and low-stakes consultations with manageable risks.

Table of Contents

1. Introduction

❏ 1.1. Context

When Dutch municipalities make decisions impacting their citizens, they can hold consultations in the form of, e.g., opinion polls. In consultations, citizens can voice their opinions, wishes, and how they want to be involved in the consultation's topic. Consultations thus allow the municipality to gain valuable feedback and (political) support from the citizens regarding a local decision impacting their citizens.

Nevertheless, consultations have such a low voter turnout that their results cannot be deemed representative of their voters (Aarts, 1999; Lijphart, 1998). Voters find it too troublesome to go to the polling station for every consultation which they consider not relevant or important enough. However, they would likely cast their vote digitally if this can be done easily (Aarts, 1999). It has been shown that voter turnout of absent voters and occasional voters can be increased with the usage of electronic voting (Petitpas et al., 2020).

IRMA vote[1] is a project of the Radboud University[2] and supports electronic voting. **Electronic voting** uses electronic means to assist or entirely perform the casting and counting of votes (Buchsbaum, 2004). **Remote electronic voting** is a form of electronic voting in which a voter can cast their vote remotely from any location, using the internet, instead of from a physical voting booth location (Krimmer et al., 2019).

❗ It might be possible to safely increase accessibility and voter turnout of consultations and decrease cost by using remote electronic voting platforms that use a **split server design**. The core of this design is splitting the consultation process over two separately hosted servers. The first server checks eligibility (and therefore does know the voter's identity), while the second server only registers the cast votes (and therefore does not know the voter's identity). In this research, we will look into the details of such a system that uses the split server design, called **IRMA vote**.

❏ 1.2. Goal

This research aims to increase the cybersecurity of remote electronic voting in consultations. We do so by mapping out all possible attack scenarios in split server electronic voting processes and performing a risk assessment for each attack scenario. Based on the risk assessment, we can determine whether remote electronic voting platforms using the split server design are secure.

[1] https://privacybydesign.foundation/irma-explanation/ For more information on IRMA.

[2] https://www.ru.nl/ihub/research/research-projects/irma-vote

By showing that remote electronic voting systems using the split server design in consultations are relatively safe, we demonstrate that such systems are a viable option for consultations. Consequently, voter turnout to such consultations might be increased. When access and voter turnout to consultations are increased, this might result in the consultation depicting a more representative public opinion, having more public support, and having more public trust. In the end, remote electronic voting might, in this manner, benefit the democratic process.

❏ 1.3. Scope

The scope of this research is remote electronic voting for small-scale and low-stake consultations (see section 3.5) via devices connected to the **internet**. Therefore, voters can submit their votes electronically from any location with the use of the internet (Zissis & Lekkas, 2011).

We will only consider that voters use the following devices in a remote electronic voting platform for consultations: computers (laptops and PCs), smartphones, and tablets. Furthermore, all other possible voting methods, such as casting a vote through a phone call, are excluded.

There are a multitude of cyberattacks possible in an electronic voting process. For the remote electronic voting platform, we define the scope of their possible attack scenarios with respect to the servers, the connection between the client and the server, and the interface, i.e., the browser. Therefore, the possible attack scenarios include attacks against both the **back-end** and the **front-end** of the remote electronic voting system.

In this research, we do not focus in-depth on the technical side of cybersecurity. The focus lies on the organizational side of cybersecurity, and on analyzing what range of cyberattacks is possible for remote electronic voting platforms using the split server design in consultations.

❏ 1.4. Contribution

This research will offer insights into the possible attack scenarios, the resulting risks, and safety of a remote electronic voting platform using the split server design in consultations. The attack scenarios and corresponding risks can be used as a framework to determine which steps and processes need to be taken when, where, and by whom to achieve cybersecurity within remote electronic voting consultations using the split server design.

❏ 1.5. Method

In this research, we try to determine the level of risk for remote electronic voting platforms using the split server design given plausible cyberattack scenarios in low-stake and small-scale consultations. In order to determine this, we will first design relevant and plausible cyberattack scenarios based on security requirements, threats, and attacker models. Secondly, based on the impact of each threat and the likelihood of each attack scenario, we will perform a risk assessment. Relevant literature and sources concerning remote electronic voting will be used to identify and design the security requirements, threats, attacker models, attack scenarios, and risks.

❏ 1.6. Reading guide

We use the below defined structure for risk assessment regarding cybersecurity in this research. First, the relevant security requirements are determined for the relevant system (chapter 4). The relevant system is all remote electronic voting platforms using the split server design such as **IRMA vote** (chapter 5). The security requirements are used to determine threats to the system (chapter 6). The security requirements and threats are then used together with the attacker model (chapter 7) to determine relevant attacks against the system, i.e., attack scenarios (chapter 8). In the end, risk will be assessed for each attack scenario (chapter 9), completing the risk assessment.

Finally, we discuss the results of the risk assessment for remote electronic voting platforms using the split server design in low-stakes and small-scale consultations (chapter 10), suggest ideas for future research (chapter 11), and draw conclusions (chapter 12).

2. Background Literature

In this chapter, general background information on electronic voting will be given.

❏ 2.1. Background on electronic voting

Motivations for implementing electronic voting methods are often based on the possibility of more accurate results, quicker results, reduced overall cost, and increased accessibility (Okediran et al., 2011). Motivations to not use electronic voting methods are often based on the cybersecurity risks inherent to electronic voting methods and a possible digital divide. The digital divide entails that certain groups of people will be excluded in electronic voting processes due to the (technical) requirements necessary to vote.

It is good to keep in mind that all voting processes consist of **three phases**: the casting of the vote, the processing of the vote, and the counting of the vote.

Electronic voting methods and related research have gone through quite some commotion and developments. The following paragraphs will provide some background on these commotions and developments.

❏ 2.2. Current state of research on electronic voting

According to Jonker et al. (2013), most of the research in electronic voting is done on privacy-related topics and verification methods. In the last few years, the amount of research done on this topic has increased due to digitalization and an increase in worldwide adaptations. The focus has also shifted towards remote electronic voting (Krimmer et al., 2019). However, the **consensus** remains that electronic voting methods have universal struggles with security issues, transparency issues, verifiability issues, and a general lack of public trust in the electronic systems (Trechsel et al., 2016; Krimmer et al., 2007, Verbij, 2014). Current research, therefore, finds electronic voting methods too risky for usage in elections and does not recommend them.

There are quite some software possibilities available that can be used for remote electronic voting. Plenty of in-depth scientific research is also published on remote electronic voting software. The conclusions from this research can be used to identify and design threats, attacker models, attack scenarios, and risks for remote electronic voting platforms using the split server design. There is still barely any research conducted on the cybersecurity side of IRMA's remote electronic voting application. At the moment, Doesburg's (2020) bachelor thesis on cryptographic schemes for IRMA vote is the only published research on the topic.

❑ 2.3. Remote voting opposed to voting on location

Remote voting directly opposes voting on location (Krimmer et al., 2019). Remote voting is done via methods like postal voting or through internet-based applications facilitating the voting process. Voting on location is done on paper in a physical voting booth at a polling station or via an electronic voting machine present at the polling station. The most significant difference between the two is whether a 'commission,' e.g., an electoral official, can supervise the act of voting. In other words, whether the environment in which the vote is cast can be controlled or not. In remote voting, supervision of the process by a commission is often impossible or not entirely possible. In remote voting, the environment is not controlled by a commission, contrary to voting on location. Requirements like secrecy and liberty can therefore not be guaranteed when casting a vote remotely (see **chapter 4**).

Remote voting and voting on location can be used concurrently in an election. One method of voting, therefore, does not exclude another (Krimmer et al., 2019). For example, postal voting (remote voting) is often used together with paper-based voting at a polling station (voting on location) to allow abroad residents of a country to participate in elections. Remote electronic voting can also be used together with paper-based voting, such as Estonia does (Springall et al., 2014). However, when it is possible to vote via 'multiple channels,' it is necessary to ensure that one individual cannot vote in different channels and cast multiple votes (Krimmer et al., 2019).

In summary, it is necessary to have multiple voting methods to provide everyone with the opportunity to participate in the democratic process (Krimmer et al., 2019). For the time being, not every citizen in a country has access or the required skill for remote electronic voting. Hence, even if remote electronic voting would be used more frequently in democratic processes, it should function as an alternative to paper-based voting and not as a replacement.

Lastly, many other differences between all forms of voting exist. There are differences regarding (cyber) security, costs, voter turnout, etc. We will highlight some of these differences in the rest of this research.

❑ 2.4. Trust in electronic voting opposed to paper-based voting

The process of voting is based on trust (Pieters, 2008). Voters vote with a certain amount of trust in the political system and the government. This trust also entails having a certain amount of trust in the election procedures. Lack of trust in the political system results in a lack of trust in the election procedures and vice versa. With digital voting systems, voters must have trust that their vote is handled correctly, e.g., kept secret, by the system and trust that the votes are counted correctly.

One of the issues of electronic voting methods is that much trust is centered around a few components, e.g., around the servers (Willemson, 2018). Of course, paper-based voting also relies on trust. Trust is, for example, placed in people correctly counting the votes or that bags of votes do not suddenly disappear during transport. The difference is that in paper-based voting, the trust is not centered around a few components but around thousands of components, e.g., thousands of counters and multiple transporters. Since there are only a few components in electronic voting, attackers have self-evident targets to attack. When the attackers are successful in their cyberattacks, the effects are also more quickly significant. Compromising one server can corrupt all votes. However, compromising one individual counter in paper-based voting does not necessarily corrupt the entire voting process. Electronic voting methods thereby suffer more from a successful attack than paper-based voting does.

In electronic voting, voters lack digital technical knowledge and have to place implicit trust in a few digital components (Willemson, 2018). Voters also have to rely and trust heavily on an experts' confidence in a digital voter system's security due to their lack of digital knowledge. On the contrary, the paper-based security process is often understandable for most voters. Most voters can understand when a paper-based process is safe and executed with integrity and when it is violated.

Another problem that often occurs with electronic voting systems is that they are also hard to audit due to how such systems are often designed, e.g., they lack independent audit trails (Jefferson et al., 2004). As a consequence of these problems, voters quickly abandon their trust in the results of a voting process when an attacker claims that they have conducted a successful cyberattack (Willemson, 2018). When voters lose their trust in the results, the electronic voting process results are (often) deemed illegitimate.

It is thus self-evident that trust in electronic voting methods can be easily broken. After all, all a malicious attacker needs to do is fabricate a plausible claim regarding a cyberattack (Willemson, 2018). Therefore, a successful cyberattack is not even needed for an attacker who only wants the voting result to be deemed illegitimate.

❑ 2.5. The Netherlands

Currently, voting in the Netherlands still happens by casting a vote in a polling station's physical voting booth ("Stemmen tellen met de hand", 2017; Verbij, 2014). If a Dutch citizen cannot physically visit the polling station due to working or living abroad, they can vote through the post.

Electronic voting methods for casting a vote are currently not used in Dutch election processes. This does not mean that there are no electronic methods and devices used in the election processes in the Netherlands. Often a variety of electronic methods are used to support the election processes. Results are counted with software help, and municipalities transfer their results via a phone call to the electoral council ("Stemmen tellen met de hand", 2017).

Electronic voting machines on location were widely used in the 1990s in elections but were banned in 2008 due to a pressure group campaign (Jacobs & Pieters, 2009). Reasons for the ban entailed easy hardware replacement opportunities by attackers, a lack of verifiability of the system due to code secrecy, eavesdropping possibilities through reading radio emissions from the devices, i.e., tempest attacks, and physical security issues with the storage facilities of the machines.

Reading radio emissions was the most significant reason for the ban since this directly compromised the secrecy requirement (Jacobs & Pieters, 2009). The secrecy requirement is often seen as one of the most important requirements in elections. Accordingly, the secrecy requirement cannot be broken in any sense. Nevertheless, this was the case with the possibility of reading the radio emissions. Reading radio emissions allowed attackers to read what someone was voting in real-time from a distance which breaks the secrecy requirement.

In the Netherlands, electronic counting of paper cast votes was not possible since the ballots were too huge to feed automatically into a machine (Jacobs & Pieters, 2009). As of the last election in 2017, manual counting of votes is still the default, and supporting software for counting is prohibited from being connected to the internet due to cyberattack concerns ("Stemmen tellen met de hand", 2017).

Several remote electronic voting trials have been conducted in the Netherlands, notably for water board elections and European elections (Jacobs & Pieters, 2009). The trials were deemed unsuccessful due to fundamental design issues and practical issues. Moreover, the trust in the systems was too low for a broader adaptation due to a lack of verifiability of the remote electronic voting process.

According to Verbij (2014), the current debate in the Netherlands about electronic voting is focused on security implications. Opponents find the electoral processes too vulnerable and essential for an electronic voting process. Advocates believe that electronic voting processes are not any different in safety from electronic banking processes when correctly implemented. The consensus on electronic banking processes is that they are widely considered safe.

Although electronic voting is at the moment not used anymore in the Netherlands, the Dutch government has indicated on multiple occasions in the past years that it is 'closely' following the developments within the field of electronic voting (Jacobs & Pieters, 2009; "Stemmen tellen met de hand", 2017). The Dutch government also stated that they are open to possible future implementations of electronic voting systems. Nevertheless, the consensus is that electronic voting systems must demonstrate sufficient security measures and generate enough public trust to even consider implementation.

In recent years, there has been a resurgence in the idea of using electronic voting methods within the Netherlands. Mostly, municipalities have been playing with the idea of possible implementations of electronic voting variations to increase voter turnout (Doesburg, 2020; Verbij, 2014). There have also been proposals for reintroducing electronic voting methods in national elections, especially regarding automatic counting ("Eindrapport Commissie-Van Beek 'Elke stem telt'", 2013).

❏ 2.6. Estonia

Estonia is the first country worldwide to use remote electronic voting as a legally binding voting method in national elections (Pieters, 2008). Estonia's remote electronic voting system has been researched extensively in many published papers. Most studies are focused on the cybersecurity side of Estonia's remote electronic voting system.

To use remote electronic voting in Estonia, voters must install special software on their computer and identify themselves via a national electronic identity system with either a smart card put into a physical card reader or a mobile-ID (Solvak & Vassil, 2016). Mobile-ID entails typing in a string of numbers and letters, received in a text, into the identity system. In recent years, QR-codes have been introduced in Estonia as part of the national electronic identity system's identification methods for remote electronic voting (Verbij, 2014).

Votes are encrypted with a public key and signed with a private key (Heiberg & Willemson, 2014; Springall et al., 2014). The voter uses the identification methods for signing the private key. After the voting period is over, the votes are decrypted with the server's private key and anonymously counted (Heiberg & Willemson, 2014). Remote electronic voting in Estonia offers online re-voting, in which only the last cast vote is counted, as an anti-coercion measure. It is also possible to vote on paper, which cancels the electronic casted vote.

Estonian voters can digitally verify whether their vote has been correctly cast (Heiberg & Willemson, 2014). The system is, however, not end-to-end verifiable (Springall et al., 2014). A voter cannot make sure that in the end, their vote is correctly counted. Therefore, the voting process's integrity relies on its voters' trust in the honest behavior of the computers, server components, and election officials.

In the Estonian remote electronic voting process, source code publications and other types of published information which should guarantee transparency and enable any third party to verify the integrity of the results were incomplete and insufficient (Springall et al., 2014). Consequently, remote electronic voting in Estonia lacks on the fronts of transparency and verifiability.

Verbij (2014) states that there have been several incidents in the past regarding remote electronic voting in Estonia. These incidents entailed decryption complications, malware, man-in-the-middle attacks, and Distributed Denial of Service (DDOS) attacks.

Therefore, Estonia's remote electronic voting system is certainly not bulletproof. Springall et al. (2014) discovered deviations from procedures, (operational) security flaws, and numerous other attack and fraud opportunities. Springall et al. (2014) concluded that there were many viable ways to attack Estonia's remote electronic voting system successfully. They were especially anxious about state-level actor threats. The possible attacks could easily disrupt the availability of the elections, compromise the secrecy of votes, create distrust in the results' integrity, alter votes, and shift votes in favor of a particular candidate.

Springall et al. (2014) also concluded that the attacks were difficult to prevent since they were caused due to fundamental architectural choices and limitations of the remote electronic voting system. Therefore, they recommended the discontinuation of Estonia's remote electronic voting system. However, the government in Estonia was dismissive in their responses to the raised concerns and issues by Springall et al. (2014). As of 2021, the same remote electronic voting system is still used in Estonia. Changes and improvements have been made through the years. Nevertheless, it is unclear if drastic changes were made addressing the underlying fundamental problems highlighted by Springall et al. (2014). At first glance, it seems that not all of these problems are adequately addressed and solved.

❏ 2.7. Other countries

There have been numerous countries worldwide in which electronic voting methods were attempted at some point (Verbij, 2014). Nonetheless, electronic voting methods are rarely used in national elections (Krimmer et al., 2007; Trechsel et al., 2016; Verbij, 2014). Electronic voting methods tend to be short-lived pilots due to universal struggles with security issues, transparency issues, verifiability issues, and a general lack of public trust in the electronic systems (Trechsel et al., 2016; Krimmer et al., 2007, Verbij, 2014).

It is also noteworthy that many countries that tried to implement electronic voting methods suffered from cyber attacks conducted by other countries (Augoye & Tomlinson, 2018). Russia attacked Estonia's remote electronic voting based elections in 2007 with a DDOS attack. A sizable DDOS attack hit Hong Kong in their online democracy poll. Ukraine was hit with a virus that was supposed to delete votes. Previous USA elections were also supposedly manipulated by an APT-level attacker. Such a history of attacks often scares away people from implementing electronic voting schemes, especially when the democratic process is at stake.

It appears that paper-based voting remains the most secure voting method in national elections. Paper-based voting will most likely continue to be the preferred method of voting in national elections for the time being. Nevertheless, electronic voting methods are often used on a small scale to support paper-based voting, even in national elections (Trechsel et al., 2016). Electronic voting methods are also increasingly used in small-scale regional political matters worldwide (Trechsel et al., 2016; Verbij, 2014).

3. Guiding Principles

This chapter contains the guiding principles, assumptions, and definitions that form the basis for the rest of the research. In the rest of this research, we will refer to these principles, assumptions, and definitions in this chapter or implicitly assume them.

❑ 3.1. Definitions

❗ We define the following terms used in the security analysis of this research, as follows (ISO/IEC 27000, 2018; Verbij, 2014):

- ***Security requirements*** are what we want to guarantee in the remote electronic voting system.
- ***Threats*** are something bad (a violation of the security requirements) that may happen to the remote electronic voting system. More specifically, threats are **actors** that perform **actions** against an **asset** for a certain **outcome** because of a certain **motivation**.
- The ***attacker model*** entails the adversaries' **capabilities** that we try to protect the remote electronic voting system from and their **goals**. Capabilities are what attackers can do and to which parts of the system they have access.
- The security requirements, threats, and the attacker model will be used together to design the ***attack scenarios***. Attack scenarios are how an attacker tries to violate a security requirement (realize a threat).
- For each attack scenario, a level of ***risk*** will be determined to discover whether we find the remote electronic voting system **secure** enough for actual usage. We will define ***risk*** as the likelihood of an attack scenario multiplied by its impact. The ***likelihood*** is the probability of something bad (violation of the security requirements) happening. The ***impact*** will be defined as the consequences of the attack scenarios for the ***security requirements***. In other words, risk is the likelihood that a cause results in an impact.

❑ 3.2. Remote electronic voting platforms opposed to digital polls

A municipality consultation needs citizens' inputs to have representative results with enough public support and public trust. Dutch municipalities often collect consultation input digitally through polls to increase accessibility and voter turnout of consultations (van Gastel, 2021). An increase in voter turnout is likely to increase the consultation result's representativeness due to an increased voter size. The results themselves also tend to have more public trust.

The alternative to digital polls is using paper-based voting for every consultation. Paper-based voting is secure, although expensive, labor-intensive, and lacking in voter turnout (van Gastel, 2021). Digital polls are, therefore, a cheap, accessible option with most likely a higher voter turnout than paper-based voting.

The downside is that digital polls often lack security measures such as checking whether a voter is eligible to vote within a particular municipality (van Gastel, 2021). Double voting is, therefore, often possible when using digital polls.

Remote electronic voting platforms using a split server design present themselves as a safer and inexpensive alternative to digital polls, depending on the necessary cybersecurity policies (van Gastel, 2021). Such remote electronic voting platforms provide the benefit of checking eligibility and preventing double voting. Furthermore, remote electronic voting platforms using the split server design might further increase voter turnout and accessibility for Dutch municipality consultations. Consequently, such platforms might be a better alternative for consultations than digital polls are.

❏ 3.3. Doesburg’s research

Doesburg (2020), who previously researched IRMA vote, proposed a cryptographic voting scheme to secure IRMA's remote electronic voting platform. In his research, Doesburg (2020) only describes whether his voting scheme suffices all legal requirements for election systems. Doesburg (2020), therefore, only describes a risk assessment for the cryptographic scheme and not for IRMAs remote voting system as a whole. This research will build upon Doesburg's (2020) research and expand on it by performing a risk assessment on the entire system of remote electronic voting platforms similar in their design to IRMA vote. That is using a split server design.

❏ 3.4. APT

An **advanced persistent attack (APT)** is a cyber-attack conducted by a nation-state, a group that is funded by a nation-state, or a significantly sizable non-state-sponsored group (Chen et al., 2014). In an APT attack, one of these actors intrudes on a computer network and remains undetected for an extended period such that they can perform malicious actions.

❗ Remote electronic voting platforms, such as IRMA, can not be considered secure enough to fend off an APT attack. With remote electronic voting, there are environments, devices, and locations that can not be fully controlled. Consequently, we cannot eliminate the possibility of APT-like attacks on both the front-end and the back-end of the system. This research **excludes** APT-like attacks on both the front-end and the back-end of the remote electronic voting system by focusing on attack scenarios concerning '**low-stake**' and '**small-scale**' consultations. This research, therefore, does not apply to so-called '**high-stake**' and '**large-scale**' consultations.

❏ 3.5. Low stakes and high stakes definitions

To distinguish between 'low stake' and 'high stake' consultations and understand when we can exclude APT-attacks from the process, we have two define these two categories. The same holds for when a consultation can be classified as small-scale or large-scale. Without a clear definition of what these categories entail, it is undesirable to use remote electronic voting platforms. The reason being, that without clear definitions, it is unknown when the risk-reward trade-off is satisfactory enough such that remote electronic voting platforms can be 'safely' used in consultations. Lastly, without clear definitions, it can not be determined whether the assumptions made in this research are valid or not for a particular remote electronic voting case.

Consultations can be classified as ***small-scale*** or ***large-scale*** based on the size of the population that the consultation affects.

Consultations can be classified as ***low-stakes*** and ***high-stakes*** based on:

- The 'societal influence' of the consultation;
- Whether it is possible to 'reverse' the consultation;
- Cross border effects;
- The amount of 'money' involved;
- Individual effects.

Societal influence entails the impact the consultation has on the population. ***Reversibility*** of the consultation entails how simple it is to reverse the consultation's result.

Cross border effects entail whether the consultation has effects that reach beyond its intended region. The ***amount of money*** involved entails whether small or large sums of money are involved relative to the population’s size. ***Individual effect*** entails whether a consultation distributes significant power to an individual.

For consultations with a considerable 'societal influence,' or which are difficult to 'reverse,' or have an 'international effect,' or involve 'large sums of money,' or have an 'individual effect'; the likelihood of a cyber-attack drastically increases (Mote, 2000; Schryen & Rich, 2009). When one of these factors is present, the stakes are considerable enough that a cyber-attack becomes increasingly beneficial and attractive for an individual or party. The result is that a cyber-attack is increasingly likely to happen. We call such situations ***high-stake*** consultations. Due to the increasing possibility of cyber attacks in 'high-stake' consultations, it could be deemed unwise to replace paper-voting in such consultations with remote electronic voting processes.

The probability of an APT drastically increases when the stakes of the consultation get higher. APT attacks are notoriously hard to prevent and detect, resulting in a high risk for 'high-stake' consultations (Chen et al., 2014). Considering the current state of remote electronic voting platforms, the risk of a successful cyberattack in 'high-stake' consultations using remote electronic voting is too high. Remote electronic voting in 'high-stake' scenarios can therefore be regarded as unwise.

With consultations that have a 'small societal influence,' that are easy to 'reverse,' have no 'international effect,' involve 'small sums of money,' and have no 'individual effect'; the likelihood of a cyber-attack drastically decreases (Mote, 2000; Schryen & Rich, 2009). We call these ***low-stake*s** consultations. The probability of an APT also drastically decreases in 'low-stake' consultations, making remote electronic voting platforms in 'low-stake' consultations a suitable alternative. The exact border between 'low stakes' and 'high stakes' should remain a topic of debate and must be uniquely defined for the context of each consultation.

❗ This research's scope concentrates on 'low-stake' consultations and excludes APT attacks by actors like, e.g., nation-states, from the attack scenarios. In the case of a 'low-stake' consultation, remote electronic voting is deemed recommendable due to increased accessibility and limited attacker models (Doesburg, 2020). It is crucial to note that any form of elections can be classified as 'high-stake,' while 'low-stake' consultations do not involve elections. Therefore, any form of **election** is **excluded** from this research since the focus is solely on 'low stake' consultations.

❗ Lastly, as will be explained in-depth in chapter 4, the secrecy requirement and liberty requirement cannot be fully met in remote electronic voting processes due to the uncontrolled environment. These two requirements are, therefore, broken 'by default' in remote electronic voting processes. In elections, these two requirements need to be met by law ("Eindrapport Commissie-Korthals Altes 'Stemmen met vertrouwen'", 2007). That is another reason why remote electronic voting such as remote electronic voting by IRMA can not be used in 'high stake' scenarios, i.e., elections.

A clear overview of whether a municipality consultation is a 'high stake' or a 'low stake' consultation can be found in the decision tree, see appendix A.

❑ 3.6. Security goals

In general, the security requirements from chapter 4 are **security goals**. Therefore, these security requirements should be met as much as is allowed by the split server design of the remote electronic voting platforms. Nevertheless, as will be indicated in chapter 4, this is only possible to a certain degree. Besides the security requirements, there are two other security goals.

Firstly, the security goal for remote electronic voting platforms used in consultations is to possess a **high detection rate** for successful cyberattacks while not negatively influencing the security requirements from chapter 4. Successful undetected cyberattacks have the most significant negative consequences (see chapter 9). In the case of a **successfully detected** cyberattack, the impact for the voters is manageable, and the impact for the consultation is that it must be held again.

Secondly, since the consultation stakes for which the remote platforms are used in this research are low, the security goal is not to prevent fraud on a small scale. The security goal is to prevent fraud on a **scale large enough** to impact the consultation result significantly. What can be considered 'large enough' and 'significant' depends on the number of eligible voters in the consultation.

4. Security Requirements

In this chapter, we will discuss and define each of the ***security requirements*** to determine the impact of attack scenarios using remote electronic voting in chapter 8. Again, security requirements are what we want to guarantee in the remote electronic voting system. Therefore, we consider the security requirements as **positive** requirements.

In section 4.1, up to and including section 4.7, we will discuss the security requirements. In section 4.8, we will discuss the voter turnout problem and define a new security requirement, called ***voter turnout***. In section 4.10, we discuss why the secrecy and liberty requirements are by default partially infringed and violated in remote electronic voting.

❗ The security requirements, i.e., security goals used in this research are ("Eindrapport Commissie-Korthals Altes 'Stemmen met vertrouwen'", 2007):

- [S1] *Transparency*
- [S2] *Verifiability*
- [S3] *Integrity*
- [S4] *Eligibility*
- [S5] *Secrecy*
- [S6] *Unicity*
- [S7] *Accessibility*
- [S8] *Voter turnout*

The security requirements from above are derived from the "Stemmen met vertrouwen" report written by the Korthals-Altes commission that advises on the Dutch electoral process ("Eindrapport Commissie-Korthals Altes 'Stemmen met vertrouwen'", 2007). The eight requirements discussed in the report are not necessary to be adhered to by law in consultations, as is the case for elections. Nevertheless, adhering as much as possible to these eight requirements in a consultation is beneficial for increasing representativeness and creating sufficient public trust.

When applying the eight requirements to remote electronic voting, **not** all eight requirements can be met in practice ("Eindrapport Commissie-Korthals Altes' Stemmen met vertrouwen'", 2007). A balance needs to be found between the eight requirements. Often based on the context surrounding the voting process, concessions need to be made on some of the requirements. We can also state that when a voting process is more classified as 'high-stakes,' e.g., a national election, the eight requirements need to be more strictly met.

Finally, the remote electronic voting platforms must have a high degree of trust from the voters in the system's security. Without sufficient trust, remote electronic voting does not work in practice, as was discussed before in section 2.4. The trust of voters in a remote electronic voting system follows from designing a voting system that adheres to the security requirements as much as possible. If the voters' trust is broken in whether the remote electronic voting system adheres to the requirements, voters doubt the voting process's integrity and results. Trust is, therefore, an essential factor that determines whether the security requirements are met in the eyes of the voters.

❏ 4.1. SR1 Transparency

Transparency in a democratic voting process means that the process should be designed in such a manner that every voter can understand the structure of the process, how the process functions, and that the process is democratically performed ("Eindrapport Commissie-Korthals Altes 'Stemmen met vertrouwen'", 2007). Transparency also requires no secrets in a democratic voting process and that all questions must be answered. The answers to the questions must also be verifiable.

The problem with all forms of electronic voting processes, including remote electronic voting, is that most voters do not have the technical knowledge and expertise to understand how the electronic voting process works ("Eindrapport Commissie-Van Beek 'Elke stem telt'", 2013). The lack of understanding reduces the transparency requirement by default in an electronic voting setting.

The result is that the trust in an electronic voting process is often lower than in a paper-based voting process ("Eindrapport Commissie-Korthals Altes 'Stemmen met vertrouwen'", 2007). When there are doubts in the voters' minds regarding the voting process's trustworthiness due to a lack of transparency, e.g., a lack of understanding of the process, the voting process results are often deemed illegitimate by its voters (Maaten & Hall, 2008).

If steps are taken to make a voting process more transparent, these steps often directly conflict with the requirement of secrecy (and liberty) ("Eindrapport Commissie-Van Beek 'Elke stem telt'", 2013). By making a detailed log of every step in a voting process, the transparency requirement increases, but this directly sacrifices the secrecy requirement (and liberty). Furthermore, in order to have maximum secrecy in a voting process, there can be no transparency. There is thus always a trade-off between the secrecy and the transparency requirement.

❏ 4.2. SR2 Verifiability

Verifiability in a democratic voting process means that the process is objectively verifiable. ("Eindrapport Commissie-Korthals Altes' Stemmen met vertrouwen'", 2007). Verifiability entails that the voting process's outcome and all the intermediate steps can be verified. Verifiability also allows for a possible recount, the capability to verify whether a vote has been correctly recorded and that only eligible people voted at most once.

There are two types of verifiability, individual verifiability and universal verifiability (Gritzalis, 2002; Pieters, 2008). Individual verifiability is the possibility that individual voters can verify their own vote in the voting process. Universal verifiability is the possibility that any observer, i.e., election officials, parties, individual voters, etc., can verify the voting process's results.

The problem of a lack of technical knowledge and expertise also holds for the verifiability requirement. The result is that voters are not capable of verifying the electronic voting process themselves ("Eindrapport Commissie-Van Beek' Elke stem telt'", 2013). The voter cannot easily verify themselves if the vote is correctly registered and counted.

In his research, Pieters (2008) states that there is a 'trade-off' between security and verifiability in electronic voting processes. Pieters (2008) states that when the protocols security and programs' security gets more robust, the verifiability requirement's legitimacy becomes weaker and vice versa. This trade-off is essential to consider while implementing extra steps in the voting process to increase security or verifiability.

Lastly, the verifiability requirement conflicts with the secrecy requirement ("Eindrapport Commissie-Van Beek' Elke stem telt'", 2013). The more verifiable the voting process becomes, the more likely it is to trace back a specific vote to a specific voter. The verifiability requirement also conflicts with voter liberty. If a voting process becomes more verifiable, the risk of third-party influences increases, and the freedom of casting a vote decreases.

❏ 4.3. SR3 Integrity

Integrity in a democratic voting process means the outcome cannot be changed by anything other than the legitimate casting of votes ("Eindrapport Commissie-Korthals Altes 'Stemmen met vertrouwen'", 2007). Integrity is a binary requirement; it either holds or does not. The integrity requirement is violated through errors and mistakes as well as through deliberate fraud ("Eindrapport Commissie-Van Beek' Elke stem telt'", 2013). The voting process must be conducted with utmost integrity, but it is an illusion to think that mistakes can be eliminated ("Eindrapport Commissie-Korthals Altes 'Stemmen met vertrouwen'", 2007). The mistakes that are made, however, must be limited to incidents and must be controllable. A method used to check voting results' integrity in electronic voting methods, including remote electronic voting, is to do manual recounts ("Eindrapport Commissie-Van Beek' Elke stem telt'", 2013; Willemson, 2018).

❑ 4.4. SR4 Eligibility

Eligibility in a democratic voting process means that only persons eligible to vote may participate in the election. Eligibility is again a binary requirement; either a voter is eligible, or a voter is illegible ("Eindrapport Commissie-Korthals Altes' Stemmen met vertrouwen'", 2007).

In electronic voting, a voter is verified through an identification step regarding their eligibility to vote. To be verified, the electronic voter needs to register, identify and authenticate themselves (Gritzalis, 2002). In the Netherlands, eligibility is checked by a municipality. The eligibility requirement conflicts with the secrecy requirement because the identification step requires a voter to reveal their identity at some point.

❑ 4.5. SR5 Secrecy

Secrecy in a democratic voting process means that it must be impossible to link the identity of the person who casts a vote to the content of their vote ("Eindrapport Commissie-Korthals Altes' Stemmen met vertrouwen'", 2007). This is often called the 'unlinkability property' (Pieters, 2008). The process must also be designed so that a voter cannot show what they have voted for. This is often called the property of 'receipt-freeness' (Pieters, 2008). Receipt-freeness prevents voter coercion and vote-buying.

There is always a possibility of **coercion** and (in)direct influences by third parties in remote electronic voting. This is seen as a **fundamental problem** of remote electronic voting, also called the **coercion problem** (Gritzalis, 2002; Kulyk & Neumann, 2020)). After all, if a vote is cast without a polling station's supervision, it can no longer be guaranteed that the citizen votes without a third party's influence or oversight. Therefore, the secrecy requirement cannot be guaranteed in remote electronic voting processes and is considered **weak** in this research (Pieters, 2009. The secrecy requirement is considered weak and not broken due to the split server design of remote electronic voting systems providing the voter anonymity so long as they do not vote while other people are present. In the end, as stated before, the secrecy requirement conflicts with many requirements.

❑ 4.6. SR6 Unicity

Unicity in a democratic voting process means that each voter may cast at most one vote (for themselves). Unicity also entails that a vote may and must only be counted precisely once ("Eindrapport Commissie-Korthals Altes' Stemmen met vertrouwen'", 2007). Unicity is a binary requirement; citizens can cast a vote at most once, and the vote is counted at most once, or it is not.

To meet the requirement of unicity, the voter must identify themselves, i.e., present their identity and authenticate themselves, i.e., prove their identity (Pieters, 2008). However, the unicity requirement conflicts with the secrecy requirement here since voters perform identification and authentication steps, revealing their identity (Gritzalis, 2002).

❏ 4.7. SR7 Accessibility

Accessibility in a democratic voting process entails that voters must be given as many opportunities as possible to participate directly in the voting process ("Eindrapport Commissie-Korthals Altes' Stemmen met vertrouwen'", 2007). Accessibility also entails that all voters should have an equal chance to participate in the voting process ("Eindrapport Commissie-Van Beek' Elke stem telt'", 2013). This is called equal accessibility (Gritzalis, 2002). Equal accessibility also entails that the voting process is user-friendly and independent of a voter's education, age, and physical condition.

One of the main benefits of an electronic voting process is a possible increase in accessibility to the voting process (Aarts, 1999; "Eindrapport Commissie-Korthals Altes' Stemmen met vertrouwen'", 2007; Willemson, 2018). For example, voters suffering from certain disabilities can benefit from electronic voting processes ("Eindrapport Commissie-Van Beek' Elke stem telt'", 2013). Nevertheless, certain groups of voters can also easily be excluded from electronic voting processes ("Eindrapport Commissie-Korthals Altes' Stemmen met vertrouwen'", 2007). Digitally illiterate voters, or voters who lack access to the digital world are quickly excluded from the voting process. An often proposed solution to this problem is to keep a paper-alternative ("Eindrapport Commissie-Korthals Altes' Stemmen met vertrouwen'", 2007; "Eindrapport Commissie-Van Beek' Elke stem telt'", 2013).

Accessibility is a requirement that often conflicts with the other requirements. When increasing accessibility, there is often a clash with the secrecy (and liberty) requirement ("Eindrapport Commissie-Korthals Altes' Stemmen met vertrouwen'", 2007; "Eindrapport Commissie-Van Beek' Elke stem telt'", 2013). Increasing accessibility in remote electronic voting settings results in a vote being taken without a polling station's supervision. Without the supervision of a polling station, it can no longer be guaranteed that citizens cast a vote without any influence and that no one knows what they have voted for ("Eindrapport Commissie-Korthals Altes' Stemmen met vertrouwen'", 2007).

It is also important to note that when the liberty and secrecy requirements need to be met in an 'absolute' sense, this automatically results in an infringement on the accessibility requirement. When these requirements need to be met 'absolutely,' this excludes, for example, voters living abroad or people who have a disability resulting in them not being capable of visiting the polling station.

❏ 4.8. SR8 Voter turnout

Dutch citizens often do not vote in voting processes which they perceive as less relevant or important (Aarts, 1999).The result is a low voter turnout. This is often the case because they find it too effortful to go to the polling station and cast their vote. Too effortful can often be described with terms like conflicting schedules, inconvenient polling stations, a long waiting line at the poll station, or simply being out of town. Effortful here also describes that the accessibility was too low, resulting in the trouble being too high for them to cast their vote ("Eindrapport Commissie-Korthals Altes' Stemmen met vertrouwen'", 2007; "Eindrapport Commissie-Van Beek' Elke stem telt'", 2013). Low accessibility here means that there were no attractive alternative voting methods. Research conducted for other countries has indicated similar reasons for not voting, that it was either too effortful or that the accessibility was too low (Pieters, 2008; Verbij, 2014).

Research has indicated that remote electronic voting only works as a solution to the low voter turnout problem when publicity campaigns promote the voting process (Gibson, 2001). A lack of communication about a consultation or an election is a significant contributing factor in low voter turnout numbers in Dutch municipalities (Ostaaijen, Epskamp, & Dols, 2016). Residents are often unaware of a consultation topic and often do not know that it is possible to vote in a specific municipality consultation

Nevertheless, it has been shown that Dutch citizens are more likely to vote in low-stakes voting processes when they are capable of remote electronic voting (Aarts 1999; "Eindrapport Commissie-Korthals Altes' Stemmen met vertrouwen'", 2007). It has also been shown that voter turnout of absent voters and occasional voters can be increased via remote electronic voting (Petitpas et al., 2020). Moreover, remote electronic voting applications might also offer extra communication possibilities. Extra communication possibilities can entail sending notifications to residents to remind citizens of participating in consultations. Remote electronic voting, therefore, offers a solution to the low voter-turnout problem for consultations.

The representativeness of a voting process is dependent on voter turnout. The more citizens vote, the more likely the result is more representative. A lower voter turnout results in a less representative outcome, as the results only represent a small percentage of the citizens, i.e., the views of the minority (Aarts, 1999; Lijphart, 1998). In such a case, the majority of the population did not vote, and thus, a large group of people did not influence the voting process. Consequently, elected policies become more unrepresentative and reflect the population's wishes less. As a result, any possible decision based on the voting process is less likely to be accepted due to its lower (political) support. Municipalities face a similar scenario when dealing with low voter turnouts.

Since a comprehensive and large voter turnout is of utmost importance to a democratic voting process, we add it to the eight original requirements of the Korthals Altes report (2007) as a 'ninth requirement.' We will call this new requirement the requirement of ***voter turnout.*** This requirement is not **explicitly mentioned in the report from Korthals Altes** (2007).

❏ 4.9. (Liberty)

Liberty in a democratic voting process means that eligible voters must be able to determine their choice in freedom and must be able to cast their vote in freedom ("Eindrapport Commissie-Korthals Altes' Stemmen met vertrouwen'", 2007). Freedom here means free from influence.

For remote electronic voting, simultaneously integrating the liberty and verifiability requirement is quite a challenge (Kulyk & Neumann, 2020). If the process allows users to verify that their vote has been counted and cast correctly, it will violate the liberty requirement because voters can prove that they have voted against an adversary. Logically, this will also violate the secrecy requirement since voters can show what they have voted for.

Furthermore, the liberty requirement conflicts with the accessibility requirement because when increasing the accessibility of a voting process, the probability of third parties influencing the casting of the vote increases ("Eindrapport Commissie-Korthals Altes' Stemmen met vertrouwen'", 2007).

❗ It is crucial to understand that liberty cannot be guaranteed in remote electronic voting since the circumstances in which a voter casts their vote can never be fully controlled, due to the coercion problem (Kulyk & Neumann, 2020). Consequently, we assume that the liberty requirement from the "Stemmen met vertrouwen" report written by the Korthals-Altes commission that advises on the Dutch electoral process is **violated** in remote electronic voting processes ("Eindrapport Commissie-Korthals Altes 'Stemmen met vertrouwen'", 2007. We, therefore, leave this requirement mostly out of this research scope and don't have any expectations regarding this requirement. In other words, **we do not consider liberty as a security requirement** that remote electronic voting systems need to meet.

❏ 4.10. Lack of secrecy and liberty in remote electronic voting

❗ It is deemed impossible to comply with all requirements in an 'absolute' sense in remote electronic voting. The liberty and secrecy requirements are two requirements that are 'by default' **violated** and **infringed** in remote electronic voting due to the uncontrolled environment. Consequently, we assume that we have a "**weak**" secrecy requirement in consultations compared to the **"strong"** secrecy requirement related to Dutch elections (Pieters, 2009). We also assume that the liberty requirement is **violated** in consultations.

It is important to try to meet the secrecy (and liberty) requirement as much as possible in a remote electronic voting process. Nevertheless, since this requirement cannot be met in an absolute sense, it is crucial not to make sacrifices on the verifiability and transparency requirements to meet the secrecy (and liberty) requirement more strictly. The reason for this is that the most important thing for a representative result with significant trust is a highly transparent and verifiable remote electronic voting process. This is essential for any remote electronic voting process since the voters already suffer from a lack of (digital) technical expertise to understand the digital voting system as a whole.

The secrecy requirement (and liberty requirement) will also always be a bottleneck regarding the cybersecurity of remote electronic voting processes. It is implied that the more the secrecy requirement is met, the less detectable cyberattacks will be due to a higher degree of secrecy regarding the process. Consequently, there needs to be a balance between the secrecy requirement and the level of detection. Furthermore, a more transparent and verifiable process lessens the secrecy requirement in remote electronic voting processes by nature. A more transparent and verifiable process might also lead to new attacking opportunities for attackers due to the extra publicly available information that attackers can study and exploit.

❗ Lastly, because the security and liberty requirements are consecutively infringed and broken in remote electronic voting processes, remote electronic voting platforms are **unsuitable for any Dutch election** since the eight requirements must be upheld by law in (official) Dutch elections.

5. System Design

❗ The following chapter describes the functioning of **IRMA vote** for remote electronic voting. This chapter, therefore, also describes each of the steps taken in remote electronic voting platforms similar in their design to IRMA vote, that is using a **split server design**. Findings and information in this research, in general, apply to all remote electronic voting platforms using the split server design in low-stakes and small-scale consultations, such as IRMA vote.

❏ 5.1. IRMA

The application used in this research as an 'example case' of a split server design to support remote electronic voting in consultation is IRMA vote ("Privacy by Design Foundation", 2021). IRMA[3] is a so-called attribute-based authentication system developed by the 'Privacy by Design' foundation. IRMA allows its users to reveal certain attributes about themselves to a verifier. Attributes are pieces of information of an individual revealing whether someone, for example, is a student or is of legal drinking age. IRMA vote is an application developed by IRMA and can be seen as a flexible, privacy-friendly, and secure application for remote electronic voting.

Revealing one's attributes does not necessarily entail revealing one's identity since attributes do not have to be uniquely identifying, i.e., ***non-identifying attributes***. Non-identifying attributes are anonymous since they do not disclose who is involved and can apply to multiple people, e.g., someone's age. However, some attributes are ***uniquely identifying***, e.g., a BSN number. The IRMA attributes are carried by the IRMA application itself and allows its users to gain authorization to processes by selectively disclosing their attributes.

The IRMA remote electronic voting process consists of roughly three steps[4]: registration, casting a vote, and verification (Botros et al., 2021). IRMA's remote electronic voting process tries to meet the eight requirements as much as is reasonably possible ("Irma-demo.stemmen.stempas - IRMA credentials", 2021; "IRMA-vote", 2021; "Raadplegingen met IRMA", 2021).

[3] https://irma.app/docs/what-is-irma/ For an in-depth explanation of the ten steps involved in a typical IRMA voting process.
[4] https://election-register.sustainablesoftware.info/ A demo for using IRMA in municipality consultations.

❏ 5.2. One voting phase

In practice, the municipalities who were willing to work with IRMA have informed IRMA that a separation in time between voter registration and vote casting is unworkable (Botros et al., 2021). The extra effort required of the voters will compromise voter turnout too much with these two separate phases. It also introduces the risk that voters forget to register or cast their vote. For this reason, there are at the moment **no two separate phases** between voter registration and vote casting. The reduction of two phases to one results in additional security risks of an attacker being able to easier link a voter's identity to their casted vote through time-based attacks, compromising the secrecy requirement.

Doesburg (2020) proposes an obvious solution to solve this cybersecurity risk by implementing two distinct phases: voter registration and vote casting. These phases are performed at different points in time to ensure the secrecy requirement. Nevertheless, this solution was deemed unworkable by the municipalities and therefore was not implemented. Accordingly, an alternative measure to solve this cybersecurity risk was implemented, which entailed splitting the voter registration and vote casting over two separate hosted servers.

❏ 5.3. IRMA-based remote electronic voting

In all three of the following steps in IRMA-based remote electronic voting, **QR codes** are scanned with the IRMA application on a mobile device to complete and confirm the corresponding action.

In the first step of an IRMA-based remote electronic voting process, the voter **registers** themselves to a registration website of the particular voting process (e.g., the municipality in a consultation), called **server A** (Botros et al., 2021). The voter **identifies** themselves in this step and proves *eligibility* ("Irma-demo.stemmen.stempas - IRMA credentials", 2021; "IRMA-vote", 2021; "Raadplegingen met IRMA", 2021). Whether a voter is eligible to vote is determined by an allowed list containing relevant attributes, such as names. This allowed list is configured to server A by an admin (Botros et al., 2021). Subsequently, server A issues a random blind voting number to the voting card of the voter (Botros et al., 2021; "Irma-demo.stemmen.stempas - IRMA credentials", 2021; "IRMA-vote", 2021; "Raadplegingen met IRMA", 2021). Server A does not know the random blindly issued voting number but does know that it has issued a voting card to a particular voter. The result is that the voter is anonymous, guaranteeing *secrecy* (and liberty). The voting card is in principle issued only once, preventing double voting, which guarantees *unicity*.

The second step is the **casting of the vote** (Botros et al., 2021; "Irma-demo.stemmen.stempas - IRMA credentials", 2021; "IRMA-vote", 2021; "Raadplegingen met IRMA", 2021). The voter casts their vote at a separate voting website from server A, called **server B**. Server B does not know the voter's identity and only knows that the voter is eligible to vote through their voting card, guaranteeing *secrecy* and *eligibility (*and liberty).

The voter confirms their choice by signing it with the random blindly issued voting number. This guarantees *integrity* and authenticity. The voting number is registered on server B when casting a vote to prevent double voting and to guarantee *unicity*. In other words, an individual cannot cast a second vote with a voting number that is already used.

Furthermore, in this second step, a voter receives **proof** that they have cast a vote on server B. This proof can be used together with the voting card by a voter to prove that their vote appears to be deleted from the public voting register while they had voted. The proof and the voting card can then be used by the party holding the voting process to *verify* whether the voter is speaking the truth. In summary, the voter can prove to the party organizing the voting process that an attacker has deleted their vote. The proof, therefore, serves as a **non-repudiation** property.

In the third step, all casted votes from server B are transformed to a third separate server from server A and server B, called **Server C** (Botros et al., 2021; "Irma-demo.stemmen.stempas - IRMA credentials", 2021; "IRMA-vote", 2021; "Raadplegingen met IRMA", 2021). Server C publishes all casted votes (the voting choices and the voting numbers) in a ***public voting record*** such that everyone can verify the results, guaranteeing *verifiability* and *transparency*. Consequently, each individual can verify that their vote was correctly cast and contributed to the result by looking up and checking their voting number. Every third party can also use the public voting record to recount the votes.

Furthermore, revoking voting cards or changing and deleting one's vote is currently **not** used in IRMA's remote electronic voting system. Finally, there is no electoral council as an actor in such remote electronic voting processes. Such remote electronic voting processes are, after all, not an official election according to the electoral law. There is, however, an ***electoral official*** in IRMA-based remote electronic voting who supervises the process and, therefore, should verify the process (Botros et al., 2021; Doesburg, 2020). This electoral official signs the public voting record with a public key.

- The process is also visualized in an overview diagram which can be found in appendix B.

❑ 5.4. Step-by-step overview of IRMA based remote electronic voting in municipality consultations

Remote electronic voting platforms similar in their design to IRMA vote, consist of the following steps when used in Dutch municipality consultations ((Botros et al., 2021; Doesburg, 2020; "Irma-demo.stemmen.stempas - IRMA credentials", 2021; "IRMA-vote", 2021):

1. The identification of the voter to the municipality on **server A** through the browser website.
2. The municipality verifies the user and checks *eligibility*.

3. The municipality issues a random and blindly issued voting number from **server A** to the user's IRMA voting card on their IRMA application. This guarantees *secrecy* (and liberty).
4. The voter casts a vote in the voters interface through a separate browser website on **server B**. The cast vote is signed with the anonymous IRMA voting card, i.e., the random blindly issued voting number on a seperate **server B**. This guarantees *integrity* and authenticity.
5. The recording of the vote on **server B.**
6. The voting number is also registered on **server B**. This guarantees *unicity*, i.e., no double voting. The voter also receives the proof that they have voted on **server B**.
7. The votes are counted on **server B.**
8. The votes are manually downloaded and transformed to a separate server, **server C.**
9. The results of the local consultation from **server C** are publicly published. This guarantees *verifiability* and *transparency***.**
10. The process also includes the possibility for any interested party to recount and verify the published results' votes through the publicly published results. This guarantees *verifiability* and *transparency*.

❗ The tracking of voters must be avoided. Therefore, voting cards must be distributed separately from the registration of the votes on two separate and distinct servers, server A and server B. In order to verify the process safely and to guarantee secrecy, the votes are published through a separate and distinct server C. The **same** party, the municipality, hosts server A and server B in the case of **IRMA vote**. A different party from the municipality hosts server C. In other words, server A and server B are hosted on behalf of the municipality by different hosting parties, while server C is completely separately hosted by another party than the municipality. Lastly, there are firewalls placed between all three servers, which disallows all communication.

- The process is also **visualized** in a flow diagram which can be found in appendix C

❑ 5.5. Using signing to cast votes

In remote electronic voting processes similar to IRMA vote, the voter must first authenticate itself to server A, i.e., the municipality, in a consultation ("Randomblind issuance IRMA docs", 2021). Server A determines whether the voter is eligible to vote. In this process, Server A only learns the minimal necessary information about the voter's identity. Server A issues a voting card to the voter if the voter is eligible to vote. The voting card contains a signature, the random blindly issued voting number. In this process, server A does not discover the value of the random blindly issued voting number. Consequently, server A cannot learn the content of the voter's vote. Additionally, in this process, a voter must also be uniquely identified by server A to prevent double voting.

The **signature** is a digital signature, including a specific attribute ("Technical overview IRMA docs", 2021a). A voter signs their vote with the signature, i.e., the random blindly issued voting number ("Randomblind issuance IRMA docs", 2021). The digital signature ensures the properties of authentication, non-repudiation, the integrity of the signed message, and unforgeability. The digital signature is also cryptographically verifiable. The attribute-based signature signs a message, i.e., its (SHA256) hash with a private key, which is technically similar to disclosure proofs[5]. Consequently, the signature allows a voter to reveal only certain information about themselves. The signature allows the voter to show that they are eligible to vote without revealing their identity in remote electronic voting.

The voter's signature, the voter's decision, and the voter's attribute, i.e., the random blindly issued voting number, are stored in a database at server B ("Randomblind issuance IRMA docs", 2021). Server B consequently only knows what an individual voted for and that the voter was eligible to vote, but the voter's identity is not revealed to them at any point in the process.

The database of server C, i.e., the server's contents with the votes, is published after the remote electronic voting process ("Randomblind issuance IRMA docs", 2021). Every voter can now verify whether their vote was correctly cast by checking if their attribute-based signature, i.e., the random blindly issued voting number, is among the published votes. Furthermore, since the public key is known, any party can check whether the voting process's result is correct by simply recounting the published votes.

[5] https://irma.app/docs/overview/#attribute-based-signatures For an in-depth technical explanation.

6. Threats

As with all electronic processes, remote electronic voting is vulnerable to cyberattacks resulting in a loss of confidentiality, integrity, and availability of information (Zissis & Lekkas, 2011). A cyberattack can have as a consequence that the voting results are deemed illegitimate. However, only if a cyberattack is **detected** can it be determined if the results are illegitimate. If a cyberattack goes unnoticed, the voting process might, consequently, have incorrect and illegitimate consequences. The impact of such illegitimate and incorrect consequences might be small or big, depending on the content of the voting process.

In order to design attack scenarios, we have to model the relevant **threats**, i.e., something bad (a violation of the security requirements) that may happen to the remote electronic voting system. More specifically, threats are **actors** that perform **actions** against an **asset** with a certain **outcome** due to a specific **motivation**. This chapter will identify and describe the most relevant threats and their impacts on remote electronic voting platforms.

Each of the following T's will describe a threat in general. Each sub-component of a threat, e.g., T1a, will give a more specific instance of that particular threat.

- **T1.** **Breaking voting secrecy**
  - **T1a** Leaking the content of the votes
  - **T1b** Family voting

- **T2.** **Voting fraud, i.e., voting as someone else**

- **T3.** **Disrupting the voting process**
  - **T3a** DDOS attacks
  - **T3b** Deleting server data

- **T4.** **Changing the results of the voting process**
  - **T4a** Deleting votes
  - **T4b** Adding votes
  - **T4c** Changing votes

- **T5.** **Breaking trust**
  - **T5a** Deleting, leaking or changing cryptographic evidence

❏ 6.1. Breaking voting secrecy (attack on confidentiality)

T1. An attacker breaks voting *secrecy* through reading and leaking what a particular voter has cast as their vote.

Threats that break voting secrecy can have far-reaching consequences. The *secrecy* (and liberty) requirement is meant to protect the voter from the political system forcing the voter to vote in a specific manner (Pieters, 2008). When a cyberattack compromises a voter's secrecy (and liberty), not only the trust of the voter in the political system and the election procedures are broken, it also may have social, economic, and political consequences for the individual voter. In extreme cases, a leak might result in an individual voter suffering physical harm caused by a political adversary unsupportive of their particular voting choice.

T1a. The threat in this scenario is an attacker becoming capable of reading and leaking the **content of the votes.** The content of the votes is the link between the chosen voting option and the voter's identity. Consequently, such an attack breaks the *secrecy* requirement. In the end, the result of the voting process can be deemed illegitimate when an attack breaks voter secrecy. Additionally, the voting process must be held again.

When an attacker can read the content of the votes, the attacker also breaks voter liberty. If any person can read the content of a vote while also knowing the voter's identity, a voter can no longer cast their vote free from any influence. Consequently, a voter can be pressured into voting for a particular option, and thus, voter coercion and vote-buying may happen.

T1b. ***Family voting*** is a form of voting coercion in which voters are forced to give up their *voting secrecy* and liberty due to social or physical pressure, often from family members or close friends (Pieters, 2008; Verbij, 2014). Family voting can only occur when individuals are voting while other people are present while casting a vote, i.e., casting a vote in an uncontrolled environment. This is an inherent problem with any remote voting method.

➔ **T1, T1a and T1b** are relevant threats for all types of voting methods.

❏ 6.2. Voting fraud, i.e., vote as someone else (attack on integrity)

T2. An attacker can commit voting fraud, i.e., vote as someone else**.** In this manner, the attacker can cast extra votes of their preference. This breaks the *unicity* and *integrity* requirements.

➔ **T2** is a relevant threat for all types of voting methods.

❏ 6.3. Disrupting the voting process (attack on availability)

T3. By disrupting the voting process in any manner, an attacker can disturb or break *any* of the security requirements. We assume that disrupting the voting process implies disrupting the usage of the remote electronic voting platform in the voting process.

T3a. Distributed Denial of Service (DDOS) attacks against the server-client communication result in eligible voters not being able to access the remote electronic voting platform (Ehringfeld et al., 2011). DDOS attacks, therefore, break the *accessibility* requirement and *voter turnout* requirement (Okediran et al., 2011; Springall et al., 2014). The voting results might be impacted due to a group of voters not being capable of voting and participating. If the group of voters who cannot vote is large enough, the voting results might be deemed illegitimate.

T3b. Deleting server data can disrupt the voting process by causing accessibility and verifiability issues. Deleting server data can result in voters not being able to access the remote electronic voting platform due to the servers malfunctioning because of the missing data. This consequently breaks the *accessibility* and *voter turnout* requirements. Deleting server data might also break the *verifiability* requirement due to voters not being able to verify whether their vote was correctly counted in the outcome due to this data being deleted.

- ➔ **T3, T3a, and T3b** are relevant threats for all types of remote electronic voting methods.

❏ 6.4. Changing the results of the voting process (attack on integrity)

T4. An attacker can change the results of the voting process by deleting votes, adding votes, or changing votes.

T4a. An attacker can exclude voters from the voting process by deleting votes, and consequently, manipulate the voting results, disrupt the voting process, and break the *integrity* and *voter turnout* requirements.

T4b. An attacker can add votes to the voting process and thereby manipulate the results to their liking and disrupt the voting process. This breaks the *unicity, integrity,* and *eligibility* requirements.

T4c. An attacker can change the voters' votes in the voting process and thereby manipulate the results to their liking and disrupt the voting process. This breaks the *unicity*, *integrity*, and *eligibility* requirements. Furthermore, changing the voters' votes usually goes hand in hand with an attacker being capable of reading the content of the votes, which thus also breaks the *secrecy* (and liberty) requirement.

- ➔ **T4, T4a, T4b, and T4c** are relevant threats for all types of voting methods.

❏ 6.5. Breaking trust

T5. The trust of voters in a remote electronic voting process follows from designing a remote electronic voting process that adheres to the security requirements.

If an attacker manages to break the voter's trust in the integrity of the voting process, voters will no longer believe that the voting process meets the *security requirements*. This is also true the other way around. If voters do not trust that the remote electronic voting process adheres to the *security requirements*, then this leads to voters doubting the voting process's integrity and results. Ultimately, trust is an essential factor that determines whether the security requirements are met in the eyes of the voters and thus whether the results of a voting process are seen as legitimate by the voters, nevertheless whether the security requirements were met in practice.

- ➔ **T5** is a relevant threat for all types of voting methods.

T5a. Changing, leaking, or deleting cryptographic evidence does not change the results of the voting process. However, it impairs the verification possibilities of the process and thereby breaks the *verifiability* requirement and the voters' trust in the voting process.

- ➔ **T5a** is a relevant threat for all types of remote electronic voting methods.

7. Attacker Model

In order to design attack scenarios, we have to describe the relevant attacker models. Therefore, this chapter will describe the attacker models for remote electronic voting platforms **using the split server design** in **small-scale** and **low-stake** consultations.

The ***attacker model*** entails the adversaries' **capabilities** that we try to protect the remote electronic voting system from and their **goals**. Capabilities are what attackers can do and to which parts of the system they have access. This chapter will also state some assumptions about the ***trusted computing base***, that is, components or people that we assume cannot be compromised for some specific security requirement(s). Furthermore, we will also declare some assumptions about the attackers and what they can and cannot do (capabilities).

We consider the following **attackers**[6] to be part of the attacker model:

- [A1] *Insider threats*
- [A2] *Mother with the baseball bat*
- [A3] *Hooligan*
- [A4] *Hacker*
- [A5] *Influencer*
- [A6] *Catfisher*

We assume the following components or people to be part of the **trusted computing base** for some security requirement(s):

- **Assumption 1.** We assume a level of trust in the organizations implementing and executing the process. This means that we assume that the organizations involved in the consultation do not conspire to manipulate the process to affect any security requirement. We also assume that all organizations involved do not have malicious intentions towards the consultation. Therefore, we assume that every organization in its **entirety** is not malicious and, as a result, negatively affects *any security requirement*. This does **not** imply that we assume that (malicious) insider threats might not happen.
- **Assumption 2.** We assume that the hardware of the mobile devices and computers used by the voter to vote is not already compromised prior to an attack and, as a result, impacts any of the *secrecy, integrity, unicity, or accessibility* requirements.

[6] These seven attacker categories are **not** necessarily unique for remote electronic voting platforms using the split server design in small-scale and low-stake consultations. Similar attacker categories can be identified in all remote electronic voting processes.

- **Assumption 3.** This research excludes attacks from malicious developers because they are too challenging to prevent. We also exclude attacks based on spreading infections through the devices used by developers to maintain and implement the remote electronic voting platforms' software. Thus, we assume that the developers and their devices cannot be compromised for *any security requirement*.
- **Assumption 4.** Revocation is not part of the voting process when using remote electronic voting platforms similar to IRMA vote. Therefore this is not a component that can be exploited by an attacker to maliciously revoke voting cards and affect the *integrity* requirement.
- **Assumption 5.** We assume that an attacker cannot exploit a voter who wants to delete or change their vote after casting a vote. Such an attack would consequently affect the *eligibility, secrecy, integrity, unicity,* and *voter turnout* requirements. The reason is that these features are not implemented as components of the voting process when using remote electronic voting platforms similar to IRMA vote.
- **Assumption 6.** We assume that the voter has established all the necessary personal details as attributes in the remote electronic voting application before voting. Therefore, we assume that the personal attributes cannot be compromised before voting for the *integrity, unicity, eligibility,* and *accessibility* requirements.
- **Assumption 7.** We make some assumptions about the security requirements themselves, see section 4.10.
- **Assumption 8.** We assume that the basic administration (Basisregistratie Personen, BRP) in which the identity of voters is stored and on which the voter's credentials are based cannot be compromised for the *eligibility* requirement.
- **Assumption 9.** We assume that voter credentials are unforgeable and cannot be compromised for the *verifiability* and *eligibility* requirements. Only the issuer (holder of the private key) can issue valid credentials ("Technical overview IRMA docs", 2021b). If a verifier receives valid attributes, it can assume that the issuer issued them.
- **Assumption 10.** We assume that server C has the property of being a write-once (append only) server and that this property cannot be compromised for the *unicity* and *integrity* requirements (Botros et al., 2021; Doesburg, 2020).
- **Assumption 12.** We assume that the installation of the remote electronic voting application is done correctly, successfully, and without complications by the voter. Therefore, we assume that the application cannot be compromised in the installation process for the *integrity, unicity, eligibility,* and *accessibility* requirements.

We also make the following general assumptions regarding **all attackers**:

- **Assumption 1.** As was discussed in section 3.4 and section 3.5, we exclude APT-level attackers from this research.
- **Assumption 2.** We assume that attackers' levels of expertise and funding are modest (Botros et al., 2021).

- **Assumption 3.** We assume that a malicious attacker has either one of the three following overall motivations to attack a remote voting consultation:
  1. An attacker who has personal opposing views against the consultation content and is closely affected by the consultation content.
  2. An attacker who is hired to attack the consultation remote voting process. Since someone with malicious intent does not have to conduct a cyberattack themselves, malicious cyberattacks can easily and cheaply be bought online as a service (Verbij, 2014). For example, conducting a DDOS attack against the entire country of Serbia would only cost 6000 euros, according to Verbij (2014).
  3. An attacker who performs the attack as a form of entertainment.
- **Assumption 4.** We assume that it will always remain possible for voters to sell their vote in remote electronic voting due to the vote being cast in an uncontrolled environment. Therefore, vote-selling (and vote-buying) cannot be stopped and will not be discussed as an attack.
- **Assumption 5.** A *hacker*, *insider,* or a *catfisher* can also attack a **single voter**. We do not consider these attacks for threat 6.1, breaking voting secrecy, threat 6.2, voting fraud, and threat 6.3, disrupting the voting process due to their low scalability.

❑ 7.1. Insider threats

We consider everyone involved in managing or maintaining some of the steps or components in the remote electronic voting consultation process an ***insider***. The insider threat is one of the biggest threats to organizations and comes from people within the organization (Augoye & Tomlinson, 2018). Insiders often already possess a certain level of access since the organization trusts them. Insiders likewise have a more profound knowledge of the system since they work with the system. Therefore, insider threats are notoriously hard to defend against due to their initial access and deeper understanding of the system. As a consequence of insider threats, any security requirement can be compromised through possible cyberattacks abusing the vulnerabilities caused by insider threats. Of course, the security requirements might also be compromised due to the insider threat performing the (cyber) attack themselves.

Insider threats indicate that not all security is technical. A practical (remote) voting system is often reliant on organizations, procedural measures, and insiders to guarantee security (Jacobs & Pieters, 2009). Therefore, many voting schemes, including consultations using remote electronic voting platforms, rely on trust in insiders to carry out certain functions (Augoye & Tomlinson, 2018; Springall et al., 2014).

In general, the **goals** of insider threats are entirely dependent on the personal motivations or financial motivations of an insider (Saxena et al., 2020). It can always happen that an insider has conflicting personal views towards the contents of a consultation. An insider might decide to attack the consultation remote electronic voting process out of **malicious** intentions in such a case. Insider threats do not always have to be **malicious**. Insider threats more frequently occur due to **negligent** and **unintentional** actions than due to malicious actions. Both physical attacks or cyberattacks can cause insider threats.

In remote electronic voting consultations, organizational procedures entail organizational policies, organizational guidelines, organizational measures, checklists, (procedural) controls, and process or audit criteria. Many cyberattack possibilities are created by insider threats failing to adhere to organizational procedures (Augoye & Tomlinson, 2018). Failing to adhere to organizational procedures happens due to human nature, human error, or because its design is fundamentally inadequate (Springall et al., 2014).

Regarding remote electronic voting, insider threats are **capable** of stealing, deleting, or manipulating different types of voting data depending on their access level. There are multiple types of insider threats possible, each with different **access** levels. We consider insider threats to either have a low access level or a high access level.

Insiders with low access levels work at any of the involved organizations without critical access rights to the voting data. Insiders with high access levels work at any of the involved organizations with critical access rights to the voting data, such as admin rights. Insider threats with a higher access level automatically increase the impact of an attack. After all, an individual with more vital access can do more significant damage.

❏ 7.2. Mother with the baseball bat

The "mother with the baseball bat" is a pseudo-person that represents the coercion problem in remote electronic voting, discussed extensively in chapter 4.

We classify the attackers representing this pseudo-person as family members, friends, neighbors, or people from the voter's immediate environment who are present and watching while the voter casts their vote. The **goal** of this type of attacker is to influence and manipulate the voting decision of a voter.

These types of attackers are **capable** of exerting social pressure on the voter to force them to vote in a particular way. In rare cases, the attacker can also use the threat of physical violence, e.g., "the baseball bat", to force the voter to vote in a particular way.

These types of attackers have **access** to the voter's voting device in ways that they can physically view what the voter chooses to vote. The attacker may also gain physical access to the voter's device through social or physical pressure to cast a vote themselves.

As discussed in chapter 4, specifically in section 4.10, we assume that the coercion problem, i.e., "the mother with the baseball bat," is an attack that cannot be prevented in remote electronic voting due to the uncontrolled environment in which a voter casts their vote.

❏ 7.3. Hooligan

The "hooligan" is a pseudo-person that represents physical attacks against the remote electronic voting process by third parties, i.e., those who are not insiders. The **goal** of this attacker is to disrupt the remote electronic voting process through a physical attack and consequently make the results illegitimate. Such attackers are **capable** of causing damage to voting data, physically stealing voting data, or deleting voting data by physically destroying the voting data.

We consider physical attacks against voting data to imply physical attacks against the servers containing the voting data. We assume that the "hooligan" attacker does not have initial **access** to the server facilities but needs to force its way into them. We exclude physical attacks against the voting devices, i.e., stealing or destroying an **individual** voter's phone, from the attack scenarios that will use this attacker model category in chapter 8. We do not consider these attacks due to their low scalability and their likelihood being entirely dependent on how an individual voter manages their phone.

❏ 7.4. Hacker

The "hacker" is a pseudo-person **capable** of performing cyber attacks against the remote electronic voting process. The **goal** of such an attacker is to manipulate or disrupt the remote electronic voting process and its results. Such attackers can obtain **access** to all parts of the remote electronic voting process that are digital.

❏ 7.5. Influencer

The "influencer" is a pseudo-person **capable** of spreading misinformation on (online) media platforms, which causes confusion and distrust among the voters in the remote electronic voting process. The **goal** of such an attacker is to break the voters' trust in the remote electronic voting process, thereby causing doubt in the results. Another goal of such an attacker is to influence voters to vote in a particular manner by spreading misinformation. However, this last goal is a natural occurrence in every voting process and will therefore be excluded from this attacker model category. The "influencer" does not have any **access** to the parts of the remote electronic voting system.

❏ 7.6. Catfisher

The "catfisher" is a pseudo-person who represents an attacker **capable** of posing themselves as another voter. The **goal** of such a voter is to manipulate the results of the voting process by adding votes of their preference, changing votes to their preference, or deleting votes that are not in line with their preferences. We only consider cyber attacks for this attacker model category since the physical versions of this attacker category are already considered under the "mother with the baseball" category. Attackers from this attacker category are considered to have **access** to the network connections of the voters.

8. Attack Scenarios

This chapter will specify attack scenarios for remote electronic voting based on the security requirements from chapter 4, the threats in chapter 6, and the attacker model from chapter 7. Again, ***attack scenarios*** are **how** an attacker may try to make something bad happen, realize a threat. In other words, how an attacker violates a security requirement.

It is good to keep in mind that in all **three phases of voting**: the casting of the vote, the processing of the vote, and the counting of the vote, cyberattacks can occur.

❑ 8.1. Breaking voting secrecy (attack on confidentiality)

■ T1. Breaking voting secrecy and T1a. leaking the content of the votes

We assume that the attacker in this attack scenario will break the voting secrecy by leaking the votes' content. The same attacks, therefore, hold for T1 and T1a.

In this attack scenario, a malicious attacker can only conduct a successful attack, i.e., reveal and leak sensitive data by **connecting** the voter's identity to a voter's ballot choice. ***Sensitive data*** entails all data that is not publicly available after the remote electronic voting process is finished. Therefore, to succeed in this attack scenario and break voting secrecy, the attacker needs the voter's identity, the voter's ballot choice, and the connection between the two.

If an attacker only reveals the voter's ballot choice without revealing their identity, the attacker achieves nothing. The attacker achieves nothing since all ballot choices will be published after the voting period is over in the public voting register when using a remote electronic voting platform similar to IRMA vote in consultations ("Irma-demo.stemmen.stempas - IRMA credentials", 2021; "IRMA-vote", 2021; "Raadplegingen met IRMA", 2021). If an attacker manages to reveal the identity of a voter, then the attacker still achieves nothing. Revealing that an individual has registered themselves for voting in a consultation is not sensitive data.

Attack 1. A *hacker* who wants to break voting secrecy by leaking the content of the votes must successfully attack all three separate servers of a remote electronic voting platform similar in their design to IRMA vote.

First, the hacker must successfully attack server A to reveal the voter's identity (Botros et al., 2021; "Irma-demo.stemmen.stempas - IRMA credentials", 2021; "IRMA-vote", 2021; "Raadplegingen met IRMA", 2021). Server A is in itself secured and separately hosted from the other steps in the process. Secondly, the hacker must successfully uncover the voting card, i.e., the issued voting number. The voting number is random and blindly issued by server A and used on server B. Server B is also secured and separately hosted from the other steps in the process. Lastly, the hacker needs to reveal the voter's choice stored on server B.

The latter is also stored on server C after the voters are transformed to server C from server B. Server C is also secured and separately hosted from server A and server B.

➔ **Attack 1 only applies to small-scale and low-stake consultations using remote electronic voting platforms similar in their design to IRMA vote, i.e., the split server design.**

Attack 2. A *hacker* can also perform attacks against the voter's device to break voting secrecy and leak the content of the votes. Attacks against user devices allow the hacker to spy ("eavesdrop") on a voter's entire voting process. In this attack, an attacker must successfully manage to circumvent the voter's device defenses, often done through cyberattacks such as malware.

Attack 3. A *catfisher* can also manipulate the voter's connection with the remote electronic voting platform server and redirect a voter to a fraudulent voting platform that resembles the real voting platform (Okediran et al., 2011). A voter can reveal their voting choice on this fraudulent voting platform, breaking the secrecy requirement. These attacks occur through, for example, malware installed through phishing emails (Augoye & Tomlinson, 2018).

Attack 4. A *hacker* can read votes and break the secrecy requirement through cyberattacks in which the attacker is present in the communication between the client and the servers (Ajish & AnilKumar, 2020). The client (voter) and the voting server are unaware of the attacker's presence in their communication. In the end, all the communication between the client (voter) and the voting server goes through the attacker.

➔ **Attacks 2, 3, and 4 apply to all types of remote electronic voting processes.**

Attack 5. Remote electronic voting platforms similar in their design to IRMA vote are **not** spread out over two phases (see section 5.2). This offers the opportunity for a *hacker* to perform correlation attacks based on time (Botros et al., 2021). In such attacks, there is an overlap between the registering phase and the voting phase. A hacker who captures the timestamps of when a voter registers themselves and casts their vote can figure out the correlation between the identity of a voter and what vote they have cast. If a voter X registers themselves at 12:05 (Monday), and later, an anonymous person votes at 12.06 (Monday), then the anonymous person will likely be voter X.

Currently, a hacker can figure out the timestamps through the external (trusted) timestamp server, which delivers the timestamps to the digital signatures. However, there are plans by IRMA to remove such timestamps from the digital signatures (Botros et al., 2021).

➔ **Attack 5 only applies to small-scale and low-stake consultations using remote electronic voting platforms similar in their design to IRMA vote, i.e., the split server design.**

Attack 6. The security of any remote electronic voting platform is based on cryptography (Doesburg, 2020; Verbij, 2014). By breaking the cryptography, a *hacker* can decrypt all votes and read all votes' content, thus breaking the secrecy requirement.

Breaking the cryptography of a remote electronic voting platform also allows an attacker to perform other CRUD[7] operations and consequently allows an attacker to break other security requirements.

➔ **Attack 6, applies to all types of remote electronic voting processes.**

Attack 7. A *hacker* can also perform correlation attacks based on the IP addresses of the voters (Botros et al., 2021). In remote electronic voting platforms similar to IRMA vote, Server A and Server B are connected to the internet. Server A can relate the IP addresses to identifying attributes, and server B can relate IP addresses to actual votes. A hacker can break voting secrecy by revealing and leaking IP addresses. A hacker can do so by attacking both server A and B, and connecting the identity and a casted vote to the same IP address, indicating who has cast a particular vote.

Another option for a hacker would be to only attack server B and expose to which vote a specific IP address belongs and then track the IP address to a particular location. Assuming that most people vote from home, it would be easy to identify who has cast a vote based on the IP address.

➔ **Attack 7 only applies to small-scale and low-stake consultations using remote electronic voting platforms similar in their design to IRMA vote, i.e., the split server design.**

Attack 8. Colluding *insider threats* pose a dangerous threat in all types of voting processes. In remote electronic voting, similar to IRMA vote, the identification step and the voting step are deliberately separated (split) across different organizations and servers to prevent, among others, attacks that break voting secrecy.

Two insiders, in which each has access to either server A or server B, can circumvent the split design through colluding, resulting in each insider having access to both servers simultaneously. Colluding insider threats can hence circumvent the procedural measures in place meant to prevent a single insider threat. Colluding insider threats with a high access level result in more straightforward attacks. If an admin from server A colludes with an admin from server B, the voting secrecy is easily broken.

[7] **CRUD** stands for Create, Read, Update, and Delete (Martin, 1983). CRUD involves four different access levels to data storage. Every single level should have a different and separate authorization level. An attacker can gain access to one of these four operations or more.

➔ **Attack 8 only applies to small-scale and low-stake consultations using remote electronic voting platforms similar in their design to IRMA vote, i.e., the split server design.**

Attack 9. A *hooligan* can physically steal voting data from server A and server B, and consequently, leak the voting data and break voting secrecy.

➔ **Attack 9, applies to all types of remote electronic voting processes.**

- **T1b. Family voting**

Attack 10. The *mother with the baseball bat* can break voting secrecy and leak the content of the votes through an attack in which she is physically present, watching, and capturing while a voter casts their vote. The mother with the baseball bat can use social pressure or physical threats to force the voter to reveal what they have voted for and thus break voter voting secrecy. The mother with the baseball bat can also look over the voter's shoulder and spy what the voter cast as their vote.

➔ **Attack 10, applies to all types of remote electronic voting processes.**

❑ 8.2. Voting fraud, i.e., voting as someone else (attack on integrity)

- **T2. Voting fraud, i.e., voting as someone else**

Attack 11. A *catfisher* can manage to modify a voter's network connection with the remote electronic voting platform. The catfisher can then manipulate the connection and redirect the voter to a fraudulent voting platform that resembles the real voting platform (Okediran et al., 2011; Verbij, 2014). A voter can reveal relevant voting information on this fraudulent voting platform.

In such a replay attack, the client (voter) authenticates to the catfisher, and the catfisher uses this to authenticate to the voting server (Okediran et al., 2011; Verbij, 2014). The client (voter) signs and casts an encrypted vote which is sent to the catfisher instead of to the voting server. The catfisher then modifies the vote to its preference and sends it to the voting server.

In summary, a catfisher uses the revealed voter's information to cast votes of their preference by performing a replay attack. The voting server's actual vote is the vote of the catfisher and not of the voter (client). The catfisher consequently commits voter fraud.

➔ **Attack 11, applies to all types of remote electronic voting processes.**

Attack 12. An *insider* that verifies the disclosures of the remote electronic voting platform similar in their design to IRMA vote can also perform replay attacks similar to those described in **attack 10** ("Technical overview IRMA docs", 2021b). In this attack, the verifier uses what they learn in the disclosures to reveal the received attributes to other verifiers. Consequently, the verifier acts as the remote electronic voting application that possesses the attributes disclosed by it. As a result, the insider can modify the votes and send them to the voting server.

- ➔ **Attack 12 only applies to small-scale and low-stake consultations using remote electronic voting platforms similar in their design to IRMA vote, i.e., the split server design.**

Attack 13. The *mother with the baseball bat* can commit voting fraud through social pressure or physical threats. She can, through these methods, force a voter to give her access to their remote electronic voting application. Consequently, she can abuse her access to the voter's remote electronic voting system to commit fraud and cast votes of her preference.

- ➔ **Attack 13, applies to all types of remote electronic voting processes.**

❑ 8.3. Disrupting the voting process (attack on availability)

■ T3. Disrupting the voting process

An extensive range of attacks can disrupt the voting process when using a remote electronic voting platform. They generally concentrate on making sensitive data unavailable, thereby denying eligible voters legitimate access to the remote electronic voting platform (Ehringfeld et al., 2011). ***Sensitive data*** here means all necessary data for the proper functioning of the remote electronic voting process. Consequently, the remote electronic voting process results might be impacted due to a group of voters not being capable of participating and thus not voting, breaking the *accessibility* and *voter turnout* requirements. If the group of voters who are not capable of voting is large enough, the voting process results might be deemed illegitimate. An attacker can consequently disrupt the voting process. Furthermore, if an attacker manages to delete voting data to disrupt the voting process, it also breaks the *verifiability* requirement. The reason being, that voting data cannot be verified anymore if it is gone.

We assume that disrupting the voting process implies disrupting the **usage** of the remote electronic voting platform in the voting process. Attackers disrupting the voting process through, for example, protests are excluded from this attack scenario.

Attack 14. A *hacker* can attack the eligibility check performed at server A to deny voters access to the remote electronic voting platform. Server A needs to be configured by an admin with the relevant attributes that determine eligibility for the particular voting process (Botros et al., 2021). The values of these selected attributes need to be present on an allowed list uploaded to server A by the admin.

A hacker can attack the relevant types of attributes and change them, or a hacker can attack and change the value of the attributes on the allowed list. Consequently, eligible voters might be denied legitimate access to the remote electronic voting platform due to their attributes failing the eligibility check on server A. An *insider* can also perform this type of attack, i.e., the relevant admin responsible for managing the attributes.

➔ **Attack 14 only applies to small-scale and low-stake consultations using remote electronic voting platforms similar in their design to IRMA vote, i.e., the split server design.**

- **T3a. DDOS attacks**

Attack 15. A *hacker* can perform DDOS attacks against voting servers used in remote electronic voting platforms. Such DDOS attacks flood the traffic of the targeted server and overload the targeted server functioning through superfluous requests (Ehringfeld et al., 2011). Consequently, eligible voters might be denied legitimate access to the remote electronic voting process because legitimate requests can not be fulfilled anymore due to the overload.

➔ **Attack 15, applies to all types of remote electronic voting processes.**

Attack 16. A *hacker* or an *insider can* influence the outcome of a voting process by blocking the access request of a specific group of voters to any of the voting servers (Ehringfeld et al., 2011). In this attack, the attacker (hacker or insider) only wants to target a specific demographic group. Such an attacker aims to exclude a particular demographic group's votes from the voting process, e.g., a particular neighborhood in a consultation (Okediran et al., 2011).

➔ **Attack 16, applies to all types of remote electronic voting processes.**

- **T3b. Deleting server data**

Attack 17. An *insider* can delete server data through malicious, negligent, or unintentional behavior.

➔ **Attack 17, applies to all types of remote electronic voting processes.**

Attack 18. A *hacker* can attack the back-end of the voting servers and delete server data. A wide-scale of cyberattacks are possible against the remote electronic voting platform's back-end, resulting in deleted server data. For example, malware installed on the back-end can cause a data loss on the servers (Augoye & Tomlinson, 2018).

➔ **Attack 18, applies to all types of remote electronic voting processes.**

Attack 19. A *hooligan* can delete server data by physically destroying the servers at which the data is stored.

➔ **Attack 19, applies to all types of remote electronic voting processes.**

❏ 8.4. Changing the results of the voting process (attack on integrity)

We assume that all attacks that can change the results of the voting process fall either under deleting votes, adding votes, or changing votes.

■ T4a. Deleting votes

Attack 20. A *hacker* can delete votes from the remote electronic voting process by successfully attacking server B (the server on which votes are cast) and deleting votes from server B. A *hacker* can do so by, for example, only letting server B count certain votes.

A hacker must delete votes from server B before the votes are transformed to server C (the public voting register server). Alternatively, the attacker also needs to successfully delete votes from server C, before the votes are published in the public voting register.

➔ **Attack 20 only applies to small-scale and low-stake consultations using remote electronic voting platforms similar in their design to IRMA vote, i.e., the split server design.**

Attack 21. A *hacker* can delete votes from the remote electronic voting process by successfully attacking server C (the public voting register server) and deleting votes from server C.

➔ **Attack 21 only applies to small-scale and low-stake consultations using remote electronic voting platforms similar in their design to IRMA vote, i.e., the split server design.**

Attack 22. An *insider* can delete votes from the remote electronic voting process by successfully attacking server B or server C and deleting votes from server B or server C.

➔ **Attack 22 only applies to small-scale and low-stake consultations using remote electronic voting platforms similar in their design to IRMA vote, i.e., the split server design.**

Attack 23. In **attack 11**, the voter is redirected to a fraudulent remote electronic voting platform. A catfisher can also use the same attack to "delete" legitimate votes. An attacker can do so by letting the voter believe that they have correctly cast their vote, while the attacker does nothing with the actual vote (Okediran et al., 2011). The vote will then not be counted in the actual voting process.

➔ **Attack 23, applies to all types of remote electronic voting processes.**

❗ **Attack 24.** An attack that focuses on deleting votes does **not** have to be **scalable**. A *hacker*, an *insider*, or a *catfisher* can also target **only one** voter and delete their vote from the remote electronic voting process (using the earlier mentioned attacks in this attack scenario). Assuming that the particular targeted voter verifies whether their vote was included and contributed to the final result by checking the public voting register, they will notice whether their vote was deleted.

An individual voter can use the obtained proof obtained from server B (see section 5.3) to prove that they have cast a vote on server B and consequently prove that their vote was deleted. In the end, deleting one vote is seen as unacceptable since it indicates that someone has tampered with the process and its results. It might also indicate more tampering that went undetected. This possibility of tampering leads to distrust in the entire remote electronic voting process and its results, which, in the end, causes the results to be declared illegitimate.

➔ **Attack 24 only applies to small-scale and low-stake consultations using remote electronic voting platforms similar in their design to IRMA vote, i.e., the split server design.**

- **T4b. Adding votes**

Attack 25. A *hacker* or an *insider* can add votes to the consultation results by issuing extra voting cards that include the random blindly issued voting number from server A to themselves. A hacker can then use these voting cards to cast extra votes. For a hacker to retrieve extra voting cards, the hacker must impersonate additional voters and add these additional voters to the allowed list configured by an admin to server A.

➔ **Attack 25 only applies to small-scale and low-stake consultations using remote electronic voting platforms similar in their design to IRMA vote, i.e., the split server design.**

Attack 26. A *hacker* or an *insider* can add votes to the remote electronic voting process through an attack in which they manage to add votes on server B or server C.

➔ **Attack 26 only applies to small-scale and low-stake consultations using remote electronic voting platforms similar in their design to IRMA vote, i.e., the split server design.**

- **T4c. Changing votes**

Attack 27. A *hacker* can try to conduct an attack in which they change votes while votes are being cast on server B.

➔ **Attack 27 only applies to small-scale and low-stake consultations using remote electronic voting platforms similar in their design to IRMA vote, i.e., the split server design.**

Attack 28. A *hacker* can try to conduct an attack in which they change the votes on server B.

➔ **Attack 28 only applies to small-scale and low-stake consultations using remote electronic voting platforms similar in their design to IRMA vote, i.e., the split server design.**

Attack 29. A *hacker* can try to change votes from server C.

➔ **Attack 29 only applies to small-scale and low-stake consultations using remote electronic voting platforms similar in their design to IRMA vote, i.e., the split server design.**

Attack 30. A *hacker* can attack the website displaying the options a voter can choose in the remote electronic voting process. In other words, a hacker can attack the front-end of server B and change the voting options on them such that the actual cast vote is different from what a voter believes it to be. .

➔ **Attack 30 only applies to small-scale and low-stake consultations using remote electronic voting platforms similar in their design to IRMA vote, i.e., the split server design.**

❏ 8.5. Breaking voter trust

■ T5. Breaking voter trust

As discussed in section 2.4, all types of remote electronic voting processes heavily depend on trust from the voters. Voters need to trust that the voting process is carried out correctly (Pieters, 2008; Willemson, 2018). Trust is incredibly fragile in remote electronic voting due to the voters lacking the digital technical knowledge to understand the system's design. Numerous attacks can unsettle the trust of the voters in a remote electronic voting process. When detected, most of the previously mentioned attacks also tend to break the voters' trust in the remote electronic voting process.

Attack 31. The *influencer* can spread misinformation about the voting process's integrity and, as a result, cause distrust of the voters in the voting process.

➔ **Attack 31, applies to all types of voting processes.**

■ T5a. Deleting, leaking, or changing cryptographic evidence

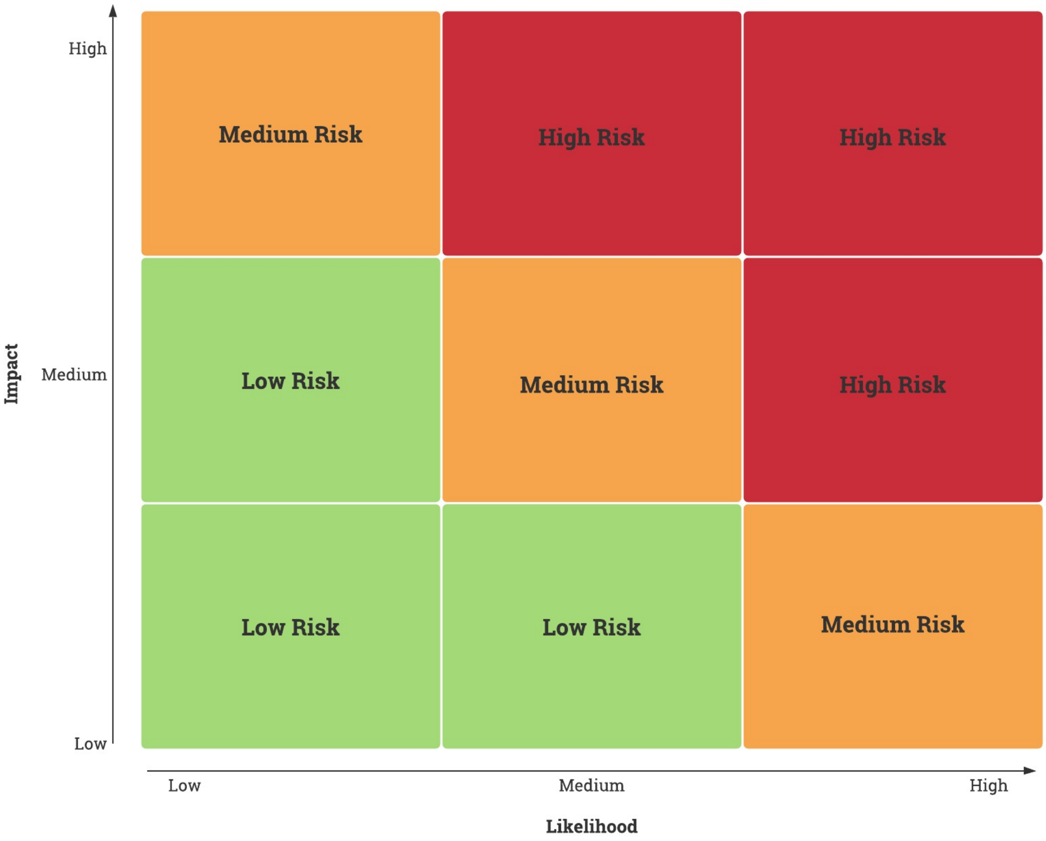
Attack 32. If any cryptographic evidence is deleted, leaked, or changed by a *hacker* or an *insider*, the entire remote electronic voting process cannot be trusted and verified anymore as being executed with integrity. As a result, the trust of the voters in the consultation has been broken.

➔ **Attack 32, applies to all types of remote electronic voting processes.**

9. Risks

❗ This chapter will define the level of risk for each attack scenario concerning remote electronic voting platforms used in **low-stakes and small-scale consultations similar in their design to IRMA vote, i.e., using the split server design.**

Earlier in section 3.1, we defined risk, likelihood, and impact. Based on these definitions, we will define the ***level of risk*** as follows (ISO/IEC 27000, 2018; Verbij, 2014):



Impact \ Likelihood	Low	Medium	High
High	Medium Risk	High Risk	High Risk
Medium	Low Risk	Medium Risk	High Risk
Low	Low Risk	Low Risk	Medium Risk

Figure 1. Depicts a matrix that determines the level of risk based on the likelihood and the impact of an attack scenario.

❑ 9.1. Breaking voting secrecy (attack on confidentiality)

■ Likelihood - T1. Breaking voting secrecy and T1a. leaking the content of the votes

Attack 1. Even if a *hacker* manages to uncover the blindly issued voting number from server B and the casted vote, the attacker still does not know to which individual voter the voting card and the casted vote belong. In remote electronic voting platforms similar in their design to IRMA vote , server B does not know the voter's identity, and server A does not know which voting number belongs to which voter (Botros et al., 2021). After all, the voting number is randomly and blindly issued (Doesburg, 2020; "Randomblind issuance IRMA docs", 2021). Server C is similar to server B since it too only knows the voter's ballot choice and blindly issued voting number, but not the voter's identity. Furthermore, server C is a **write-once (append-only) register** (Doesburg, 2020). Therefore, votes can only be written to this server, but cannot be read, without the private key.

In the case of **IRMA**, only **Server C** is operated by a **separate organization**. Server A and server B are separately hosted, but operated by the same organization, i.e., the municipality. This increases the likelihood of an attacker revealing and leaking sensitive data. For remote electronic voting platforms similar in their design to IRMA vote, all three servers are separately hosted and operated by different organizations.

In summary, due to the separation of the servers, i.e., the split design, it is **unlikely** that a hacker successfully uncovers all three components needed to reveal sensitive information, i.e., breaking voting secrecy.

Attack 2. The difficulty and likelihood of this attack depend partially on the voter and how they manage their devices' cybersecurity. Besides, a *hacker* must perform the attack separately for numerous voters to impact the consultation significantly. This attack is hence expected as too cumbersome and fruitless for an attacker and therefore **unlikely**.

Attack 3. Remote electronic voting platforms similar in their design to IRMA vote use QR-codes, which are necessary for confirmation in all steps (Botros et al., 2021; "Irma-demo.stemmen.stempas - IRMA credentials", 2021; "IRMA-vote", 2021; "Raadplegingen met IRMA", 2021). Therefore, a QR-code generated by a fraudulent voting platform from a *catfisher* is not accepted by the IRMA application, and the attack fails and is easily detected. Since the originator of the QR-code is verified by the application. This type of attack is therefore **unlikely**, under the assumption that the voter is not oblivious.

Furthermore, due to the security properties of remote electronic voting platforms similar in their design to IRMA vote, impersonation attacks are hardly possible and, therefore, **unlikely** ("Technical overview IRMA docs", 2021b). With IRMA, impersonation attacks are hardly possible because the credentials and attributes reside in the users' remote electronic voting application installation. Consequently, it is difficult for third parties to authenticate themselves as any of the IRMA users to the remote electronic voting application.

Attack 4. Due to the separation of servers, the *hacker* must be present in the client's communication (voter) with all three servers. Therefore, the attacker needs to successfully attack all three separately secured client-server connections, which makes this attack **unlikely**.

Attack 5. The voting process in remote electronic voting platforms, similar in their design to IRMA vote, is reduced to one phase (see section 5.2). The overlap between the registering and vote casting phases increases the likelihood of time-based correlation attacks (Botros et al., 2021).

Currently, the likelihood of this attack by a *hacker* is heavily dependent on the level of security the external timestamp server possesses (Botros et al., 2021). If IRMA succeeds in removing the timestamps from the digital signatures, this timestamp server would not be used anymore, removing an attacking opportunity and reducing the likelihood of this attack.

Furthermore, when there are few participating voters, which is likely in small-scale or low-stake consultations, it becomes more straightforward to deduce a person's choice if a hacker notices when this person has voted. Especially since after the consultation, all signed votes are published, including the embedded time-stamps. Considering all of this, we classify this attack as **likely**.

Attack 6. The likelihood of attacks that break the system's cryptography appears to be **low** in remote electronic voting platforms, similar in their design to IRMA vote. An in-depth analysis of the likelihood of attacks focussing on breaking the cryptography of remote electronic voting platforms similar in their design to IRMA vote is out of the scope of this research. For an extensive analysis of why we consider these attacks unlikely, see Doesburg's (2020) thesis.

Attack 7. We assume that most voters will not use an IP anonymizer to vote, increasing the likelihood of this attack (Botros et al., 2021). Furthermore, a *hacker* does not need to access the contents from both server A and server B to succeed in this attack. The IP address of a voter casting a vote and their vote themselves could be enough if a hacker manages to track the IP address to a location, which identifies the voter. Assuming that most people tend to vote from home, this attack can be classified as **likely**.

Attack 8. In the specific case of IRMA vote, the identification step and the voting step are split between two separate servers, but the same organization, the municipality, operates these two servers in consultations (see section 5.4). When using IRMA vote, the likelihood of two *colluding insiders* is therefore increased. The reasoning is that the two insiders are more likely to be familiar with each other when they work at the same organization, making them more likely to collude together.

Nevertheless, in remote voting platforms similar in their design to IRMA vote, which is not IRMA vote itself, server A and server B should **not** be operated by the same organization. This unfamiliarity between insiders helps to mitigate the colluding insider threat, making this attack **unlikely**.

Attack 9. In general, the likelihood of a *hooligan* stealing and leaking voting data is entirely dependent on how good the physical security is of the organizations operating server A and server B. Furthermore, the likelihood is dependent on the burglary skill set of the hooligan. Considering that the consultations are low-stakes and small-scale, the hooligan will most likely get caught or noticed due to a lack of such burglary skills. This high risk of getting caught makes this attack **unlikely**.

In remote electronic voting processes similar to IRMA vote, server A tends to be operated by a municipality which helps decrease the likelihood of the attack since government buildings tend to be better physically secured by their nature.

- **Likelihood - T1b. Family voting**

Attack 10. The likelihood of family voting in remote electronic voting is **high**. As stated in chapter 4, the environment is uncontrolled in remote electronic voting, making it impossible to prevent voting coercion such as family voting. This attack is especially likely in small-scale and low-stake consultations since most voters tend to know each other personally. Consequently, this makes family voting more likely since opportunities for exerting social pressure are more effective and accessible if the mother with the baseball bat knows numerous voters personally.

It is **unlikely** that the *mother with the baseball bat* uses physical threats in such consultations due to their low-stakes and small-scale. Choosing violence in a low-stake scenario seems unlikely. For example, threatening to hit someone with a baseball bat if they do not vote that the garbage gets picked up on Tuesday instead of Wednesday seems unlikely. Violence also seems unlikely, since again, people tend to know each other well personally in consultations.

- **Likelihood - Attack scenario (9.1)**

In summary, the likelihood of this attack scenario is mitigated due to the two separately hosted servers (split design) in which server A does not know the blindly issued random voting number while knowing the voter's identity (Botros et al., 2021; "Irma-demo.stemmen.stempas - IRMA credentials", 2021; "IRMA-vote", 2021; "Raadplegingen met IRMA", 2021). In contrast, server B only registers the cast votes and therefore does not know the voter's identity. Therefore, every remote electronic voting process step only knows minimal parts of either the identity, the voter's ballot choice, or the voting number. However, since voting coercion cannot be prevented in remote electronic voting, we assign a higher likelihood to this attack scenario. In summary, we can conclude that the likelihood of this attack scenario is **medium**.

- **Impact - Attack scenario (9.1)**

If an attacker manages to reveal and leak a large sum of the voter's identities, voters' ballot choices, and the link between them, i.e., breaking voting secrecy, the impact of this attack scenario would be **high**. The impact is high since such a large leak breaks the *secrecy* (and liberty) requirement of many voters, and since voter anonymity is an essential part of voting, this is highly undesired.

Breaking the secrecy requirement might result in social (or physical) consequences for the individual voters in the consultation, e.g., caused by fellow voters who are not pleased with their ballot choice. Furthermore, breaking the secrecy requirement might result in a distrust of the voters in the result of the consultation. Consequently, the entire consultation result might be deemed illegitimate when the secrecy requirement is broken on a large scale. Additionally, the consultation must be held again.

- **Risk - Attack scenario (9.1)**

Considering that the likelihood is **medium** in this attack scenario and the impact is **high**, the risk of this attack scenario can be considered **high**.

❑ 9.2. Voting fraud, i.e., voting as someone else (attack on integrity)

- **Likelihood - T2. Voting fraud, i.e., voting as someone else (attack on integrity)**

Attack 11. Remote electronic voting platforms similar in their design to IRMA vote use QR-codes, which are necessary for confirmation in all steps ("Irma-demo.stemmen.stempas - IRMA credentials", 2021; "IRMA-vote", 2021; "Raadplegingen met IRMA", 2021). Therefore, a QR-code generated by the fraudulent voting platform is not accepted by the IRMA application, and this attack fails. Therefore, this type of attack is **unlikely**, assuming that the voter checks their vote carefully before signing it (Botros et al., 2021).

Furthermore, due to the security properties of remote electronic voting platforms similar in their design to IRMA vote, replay attacks through eavesdropping are **unlikely** ("Technical overview IRMA docs", 2021b). Assuming that a verifier never uses the same random bits, i.e., **random blindly issued voting number (nonce)**, twice, an eavesdropper cannot replay a disclosure of a remote electronic voting application similar in their design to IRMA vote. The reason is that during the verification of the attributes, the verifier first sends the random blindly issued voting number (nonce) to the remote electronic voting application, and the reply of the remote electronic voting application has to fit the random blindly issued voting number (nonce) precisely. The reply contains the disclosed proofs of knowledge and disclosed attributes.

Attack 12. In remote electronic voting applications similar in their design to IRMA vote, the application never sends an entire copy of the credential's signature to the verifier ("Technical overview IRMA docs", 2021b). Parts of it are hidden using so-called proofs of knowledge. The proof of knowledge proves to the verifier that the voter knows the attributes from the credential which are not being disclosed; and that the voter owns a valid issuer signature over the disclosed and hidden attributes.

The private key is not revealed, and thus not all attributes are revealed. Consequently, even if all credential attributes were simultaneously disclosed, the verifier cannot use the received attributes and proof of knowledge to disclose the attributes themselves to the remote electronic voting platform. Together this makes this type of replay attack **unlikely**.

Attack 13. Committing fraud through social pressure cannot be prevented in remote electronic voting due to the uncontrolled environment (see section 4.10), making this type of attack **likely**. Again, this attack is especially likely in low-stakes and small-scale consultations because the voters tend to know each other well personally, making exerting social pressure more easy and effective**.**

- **Likelihood - Attack scenario (9.2)**

In summary, we can conclude that the **likelihood** of this attack scenario is low due to the design (QR-codes, nonces, and not disclosing all credentials) of remote electronic voting platforms similar to IRMA vote mitigating this attack scenario.

- **Impact - Attack scenario (9.2)**

If an attacker manages to commit voter fraud, an attacker adds or changes votes without detection. Consequently, an attacker manipulates the consultation results to their preference and breaks the *unicity* and *integrity* requirements. This is highly undesired, and therefore we consider the impact of this attack scenario to be **high**.

- **Risk - Attack scenario (9.2)**

Considering that the likelihood is low in this attack scenario and the impact is **high**, the risk of this attack scenario can be considered medium.

❏ 9.3. Disrupting the voting process (attack on availability)

- **Likelihood - T3. Disrupting the voting process (attack on availability)**

Attack 14. The likelihood of a *hacker* succeeding in this attack is dependent on the strength of server A's cybersecurity which depends on the organization managing the server. The attack is also dependent on how well the admin manages its security when creating the allowed list, configuring the attributes, and uploading the attributes to the server. In the end, we consider this attack **likely** since a hacker only needs to attack one server successfully instead of multiple servers as in the attacks as mentioned earlier.

It is **unlikely** that an *insider*, i.e., the relevant admin for configuring the attributes and the allowed list performs this attack. The reason being that the admin will be logically the first to be investigated when such an attack occurs. This makes the chance of being caught for the admin high and thus makes this attack unlikely.

- **Likelihood - T3a. DDOS attacks**

Attack 15. The likelihood of a DDOS attack performed by a *hacker* depends on the anti-DDOS measures of each of the organizations that operate the servers used for the processes in the remote electronic voting platforms. The likelihood of a DDOS attack is therefore heavily dependent on each organization.

In remote electronic voting platforms similar in their design to IRMA vote, each of the three servers can be DDOSed to deny accessibility to voters. However, DDOS prevention is harder in remote electronic voting platforms if the voting process wants or needs to adhere to specific requirements, or legislation, i.e., **the security requirements** in the case of remote electronic voting systems similar in their design to IRMA vote.

First of all, it is undesirable to blacklist IPs automatically to prevent DDOS attacks because this might impair specific requirements related to the voting process (Ehringfeld et al., 2011). Blocking all traffic from a source IP in a voting process might deprive eligible voters of their voting rights.

For example, if a university IP internet gateway is blocked in a student union election, all eligible student voters cannot vote anymore from the university IP address in the student union election (Ehringfeld et al., 2011). Blacklisting IP addresses is therefore not a desirable DDOS prevention method in remote electronic voting processes since it might hinder and block eligible voters from voting. Other DDOS prevention measures should thus be used.

Secondly, DDOS prevention measures like configuration changes, parameter adaptations, and software adaptations can invalidate the certifications and laws attached to a voting process (Ehringfeld et al., 2011). Accordingly, such DDOS prevention methods might invalidate the results and should therefore not be used.

In the end, because many standard DDOS prevention methods cannot be used in remote electronic voting platforms, we consider the **likelihood** of DDOS attacks to be **high**.

Attack 16. In small-scale and low-stake consultations, excluding particular groups by denying them access to the remote electronic voting platform through a DDOS attack is an effective method to manipulate the results to a particular preference as an attacker (*hacker* or *insider)*. For example, excluding the neighborhood that is most affected by a municipality's consultation can steer the results quickly to a particular preference.

Due to the incapability of using many standard DDOS prevention methods in remote electronic voting platforms (see **attack 15**) and the effectiveness of an attack blocking access of a particular demographic group in small-scale and low-stake consultations, we consider the **likelihood** of this attack to be **high**.

■ Likelihood - T3b. Deleting server data

Attack 17. It can always happen that an *insider* has conflicting personal views on the contents of a consultation. An insider might decide to attack the consultation remote electronic voting process out of malicious intentions in such a case. Nevertheless, since the stakes of the remote electronic voting consultations that we consider in this research are low, it is **unlikely** for an insider to have such strongly opposing views that an insider chooses to initiate an attack.

For example, an insider is unlikely to orchestrate and launch a complicated cyberattack with the risk of being caught over whether a children's playground might be placed in a particular neighborhood or not.

Negligent behavior or unintentional behavior by an insider can always happen in any voting process. Nevertheless, in remote electronic voting processes similar in their design to IRMA vote, an 'electoral official' supervises the entire process (Doesburg, 2020). We consider the electoral official enough of a regulatory force that negligent behavior and unintentional behavior are **unlikely** to happen.

Attack 18. Since the servers are operated by different and separate organizations, the **likelihood** of this attack is heavily **dependent** on how well these organizations handle their cybersecurity. We suppose that the cybersecurity of the organization operating the servers is sufficient to fend off most attacks, or else these organizations will eventually be removed from the consultation. The electoral official will also ensure that the organization maintains proper security through its regulatory oversight function.

Nonetheless, to disrupt the remote electronic voting process, a *hacker* only needs to delete data on one of the three servers successfully. This is contrary to previously discussed attacks in which the hacker needs to succeed in attacking all three servers, which makes this attack more **likely**.

Attack 19. The likelihood of a physical attack by a *hooligan* resulting in server data being deleted is partly dependent on the physical security of each of the organizations operating the servers and on the attacker's resourcefulness. Nevertheless, we consider a physical attack against the servers **unlikely** due to the consultations' small-scale and low-stakes nature. For example, an individual is unlikely to burn down an entire server facility over whether a children's playground should be placed in a neighborhood or not.

■ Likelihood - Attack scenario (9.3)

In summary, we conclude that the likelihood of this attack scenario is **high.** The reason being that many of the most effective DDOS measures cannot be used due to conflicting with the security requirements and because an attacker only needs to successfully attack one server in this attack scenario instead of multiple servers.

■ Impact - Attack scenario (9.3)

If an attack disrupts the voting process in this attack scenario, the *accessibility* and *voter turnout* requirements are primarily affected, and depending on the attack, the *verifiability* requirement might also be affected. Such attacks are always noticed and detected because we assume that voters will inform the organization managing the voting process when they could not vote. Consequently, detection results in the remote electronic voting process being held again, which mitigates the impact. Detection is, therefore, an excellent mitigation of the impact, so we consider the impact of this attack scenario **low**

■ Risk - Attack scenario (9.3)

Considering that the likelihood is **high** in this attack scenario and the impact is **low**, the risk of this attack scenario can be considered **medium**.

❑ 9.4. Changing the results of the voting process (attack on integrity)

■ Likelihood - T4a. Deleting votes

Attack 20. Remote electronic voting platforms similar in their design to IRMA vote do not have an option for voters to retract, i.e., delete, their vote in the voting process (Botros et al., 2021; "Irma-demo.stemmen.stempas - IRMA credentials", 2021; "IRMA-vote", 2021; "Raadplegingen met IRMA", 2021). This makes it more difficult for attackers to delete voters' votes since the attacker cannot exploit a pre-existing process.

Moreover, the likelihood of an attack that targets server B to delete votes is again dependent on the cybersecurity of the organization operating server B. Lastly, the attack is also time-sensitive since deleting votes from server B is only desirable when the votes are not already transformed to server C. In summary, we consider this attack **unlikely**.

Attack 21. Server C is a write-once (append-only) register (Doesburg, 2020). Therefore, once written, votes are immutable and cannot be modified or removed anymore, making a cyberattack that deletes votes from server C **unlikely**.

Attack 22. The likelihood of an *insider* threat resulting in votes being deleted is dependent on the preventive procedures organizations have in place. The procedural measures should be based on a separation and division of responsibilities, such that no trust is put into one organization or individual (Augoye & Tomlinson, 2018; Jacobs & Pieters, 2009). The objective is to ensure that problems can only occur if all involved organizations cooperate or fail somehow.

Remote electronic voting platforms similar in their design to IRMA vote should have a separation of functions between and within organizations, which, therefore, by design helps mitigate insider threats. Nevertheless, there is still an implicit trust placed in specific organizations in remote electronic voting schemes. The reason is that the organizations themselves are primarily responsible for designing, implementing, and auditing procedural measures. Therefore, the likelihood of an insider threat is heavily dependent on how well each organization designs and enforces proper procedural measures against insider threats.

In the end, we consider this attack as **unlikely** due to the reasons mentioned earlier (see attack 17) and since an insider is unlikely to conduct an attack and risk their job over a low-stakes and small-scale consultation.

Attack 23. We consider this attack as **unlikely** for the same reasons as mentioned in attack 11 (the QR code and the security properties regarding the design).

❗ **Attack 24.** **Deleting the vote of only one voter** is an effective method to disrupt a remote electronic voting process by an attacker (*hacker*, *insider*, or *catfisher*). Targeting one specific voter's vote gives an obvious target to attack against which an attacker does not have to divide its resources. It is much easier to ensure that one vote is deleted from the remote electronic voting process than doing the same for multiple votes. The reason is that an attacker does not necessarily have to use the more complicated attacks such as targeting the servers storing the votes but can also use other options, such as targeting a single voter's device. Especially for an *insider* with a high access level, i.e., an admin, it is easy to delete one vote from the servers and make it seem like an accident and thus having a low probability of getting caught.

We, therefore, consider this type of attack to be **likely**. Nevertheless, this attack's success remains dependent on whether the voters verify whether their vote has been included in the final results by checking the public voting register.

It is noteworthy that **changing** a single voter's vote to achieve a similar impact to this attack is not possible since this requires changing the digital signature and resigning the vote. In other words, this requires generating an identical random blindly issued voting number, but this is not possible due to the nonce and the split server design.

- **Likelihood - T4b. Adding votes**

Due to the nature of small-scale remote electronic voting consultations, the total voter turnout is smaller in size. Therefore, adding a handful of votes to the voting process can more easily and significantly impact the results. Consequently, an attacker only has to add a modest number of votes to manipulate the process in small-scale consultations, making detection harder and the likelihood of the following attacks higher.

Attack 25. In order to prevent an attacker (*insider* or *hacker*) from retrieving more than one voting card and thus from being able to cast multiple votes, the user is uniquely identified at **server A** against the assumed incorruptible basis administration (see attacker model assumption 8), making this type of attack **unlikely** (Botros et al., 2021; Doesburg 2020; "Randomblind issuance IRMA docs", 2021).

Impersonation is also **unlikely** in remote electronic voting platforms similar in their design to IRMA vote due to the credentials and attributes of a voter residing in the application installation of the users ("Technical overview IRMA docs", 2021b). Moreover, for an attacker to issue valid voter credentials, it needs the issuer's private key, which is not straightforward to obtain.

In summary, due to the unique identification step and the security properties of remote electronic voting applications similar in their design to IRMA vote, we consider this attack **unlikely**.

Attack 26. The random blindly generated voting number makes this attack **unlikely**. The randomly blindly voting number is issued at server A (Botros et al., 2021; "Randomblind issuance IRMA docs", 2021). An attacker (*hacker* or *insider)* needs this blindly issued voting number to add legitimate votes to server B or server C. Therefore, an attacker cannot simply add votes to server B or server C without obtaining this digital signature of information generated by server A, which is not straightforward due to the encryption. To succeed in this attack, the attacker also needs to attack server A and server B or server C successfully.

The split server design again reduces the likelihood of the attack since the attacker needs to attack multiple servers successfully instead of one, even though each attack's likelihood is again dependent on the cybersecurity of the organization operating the server.

In the end, the random blindly generated voting number, and the split server design together makes this attack **unlikely.**

- **Likelihood - T4c. Changing votes**

Attack 27. A voter uses the remote electronic voting application to confirm their chosen vote by scanning a QR-code (Botros et al., 2021; "Irma-demo.stemmen.stempas - IRMA credentials", 2021; "IRMA-vote", 2021; "Raadplegingen met IRMA", 2021). A voter will then see the actual vote, which will be sent to server B. Assuming that the voter reads and checks if this vote corresponds with their actual vote, this attack is **unlikely**.

Attack 28. Remote electronic voting platforms similar in their design to IRMA vote do not have an option for voters to change their vote in the voting process (Botros et al., 2021; "Irma-demo.stemmen.stempas - IRMA credentials", 2021; "IRMA-vote", 2021; "Raadplegingen met IRMA", 2021). This makes it more difficult for attackers to change voters' votes in the process since the attacker cannot exploit a pre-existing process.

The likelihood of an attack that targets server B intending to change votes is again dependent on the cybersecurity of the organization operating server B. Lastly, the attack is also time-sensitive since changing votes from server B is only desirable when the votes are not already transformed to server C.

Lastly, changing votes is not possible since this requires changing the digital signature and resigning the vote. In other words, this requires generating an identical random blindly issued voting number, but this is not possible due to the nonce (see attack 11) and the split server design (server A does not know the value of the random blindly generated voting number, while server B does not know the voter identity). In summary, we consider this attack **unlikely**.

Attack 29. Server C is a **write-once** (append-only) register (Doesburg, 2020). Therefore, published votes cannot be changed anymore, making an attack that changes votes from server C **unlikely**.

Attack 30. Again, a voter uses the remote electronic voting application to confirm their chosen vote by scanning a QR-code. A voter will then see the actual vote, which will be sent to server B, and will see that their chosen option differs from the actual vote. Assuming that the voter reads and checks if this vote corresponds with their actual vote, this attack is **unlikely**.

■ Likelihood - Attack scenario (9.4)

In summary, due to the design (blindly issued voting number, separation of servers, write only server, QR-codes, etc.) of remote electronic voting platforms similar to IRMA vote, any attacks that concentrate on adding, changing, or deleting votes can be deemed **unlikely**.

Furthermore, voters can always verify whether their votes have been changed or deleted through the public voting register. A voter can use the proof gained from server B and their voting card to prove that their vote is deleted. A voter can also verify whether their attribute-based signature, i.e., the random blindly issued voting number, is among the published votes and if the corresponding vote is correct. These measures detect whether votes are changed or deleted. These measures reduce the likelihood of an attacker that wants to manipulate the consultation results since a voter tends to notice when their vote has been changed or deleted, and in order to manipulate the results successfully, the attack needs to remain **undetected**. However, it is **crucial** to keep in mind that these measures also make single voter attacks effective and hence likely, since a voter can prove that their vote has been deleted through the public voting register and their obtained proof.

In the end, we conclude that the likelihood of this attack scenario is **medium.**

■ Impact - Attack scenario (9.4)

If votes get deleted, changed, or added without detection, an attacker can manipulate the consultation results to their preference and break the *verifiability, integrity, secrecy, unicity, accessibility,* and *voter turnout requirements*. This is highly undesired, and therefore we consider the impact of this attack scenario to be **high**.

■ Risk - Attack scenario (9.4)

Considering that the likelihood is **medium** in this attack scenario and the impact is **high**, the risk of this attack scenario can be considered **high**.

❏ 9.5. Breaking voter trust

■ Likelihood

Attack 31. An attack in which an *influencer* breaks trust by spreading misinformation is **likely**. The likelihood of attacks focusing on breaking the voters' trust is high because remote electronic voting processes heavily depend on trust, and trust is fragile in remote electronic voting processes (see section 2.4). The likelihood is also high since it does not require much effort and skill to fabricate a plausible attack and make voters doubt whether the voting process and results can still be trusted (Pieters, 2008; Willemson, 2018).

Attack 32. As stated before in **attack 6** (see section 9.1), attacks regarding the cryptography of the remote electronic voting platform are out of this research scope. We consider the likelihood of these types of attacks to be **low** and refer to an in-depth analysis of why this is the case to Doesburg's (2020) thesis on the topic.

■ Likelihood - Attack scenario (9.5)

In summary, we can conclude that the **likelihood** of this attack scenario is **high** due to the inherent fragility of trust in remote electronic voting, the low required effort and skill of breaking trust, and the effectiveness of breaking trust**.**

■ Impact - Attack scenario (9.5)

Breaking the voters' trust in a remote electronic voting process results in voters doubting the integrity of the remote electronic voting process, the remote electronic voting platform, the results, and the people involved in the remote electronic voting process. As discussed in section 6.5, trust determines whether the security requirements are met in the voter's eyes. Meeting the security requirements determines the legitimacy of a remote electronic voting process. Therefore without the voters' consensus that the security requirements are met, the results cannot be declared legitimate.

Furthermore, if trust is broken, voters can start doubting the remote electronic voting platform and protesting against further usage. Breaking trust in a remote electronic voting process can also result in voters doubting any future legitimate consultation results obtained through a process performed with integrity. In essence, the impact of breaking the trust of the voters is considered **high**.

■ Risk - Attack scenario (9.5)

Considering that the likelihood is **high** in this attack scenario and the impact is **high**, the risk of this attack scenario can be considered **high**.

❏ 9.6. Biggest risks

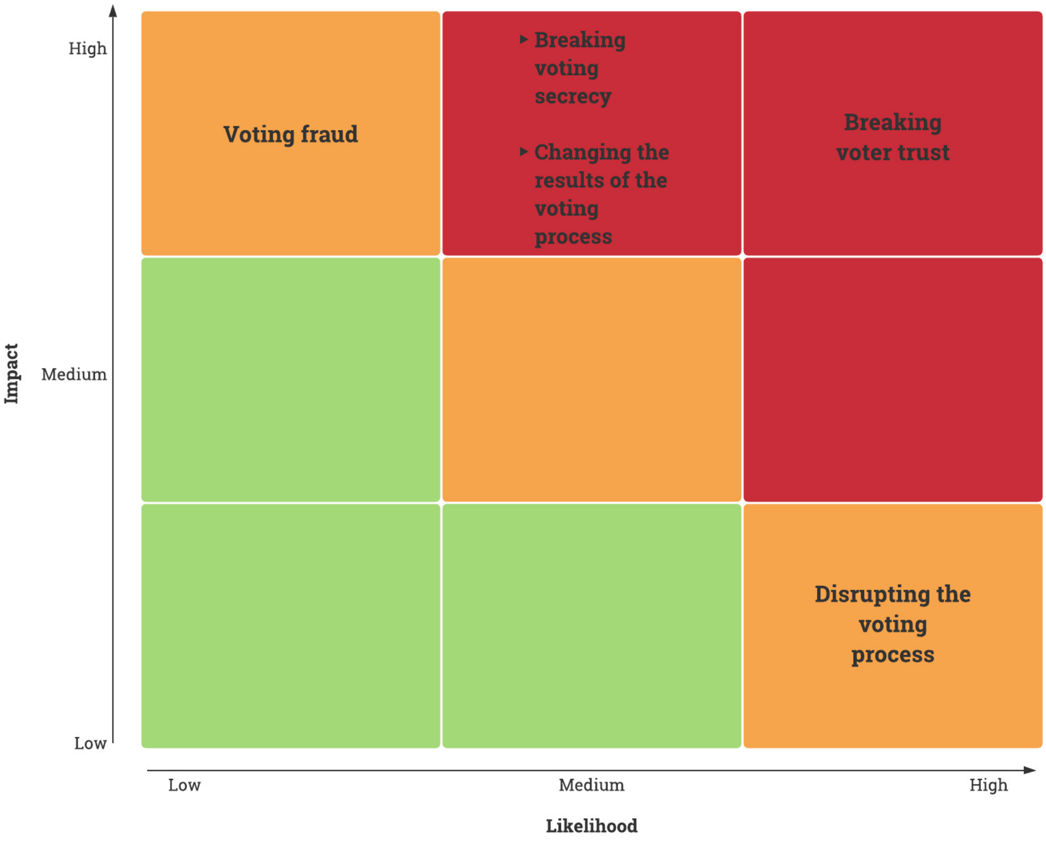


Figure 2. Depicts a matrix with the level of risk for each attack scenario.

■ Biggest risk for all remote electronic voting systems

We consider the threat of **breaking voter trust** to be the most **dangerous**, with the **highest risk** for **all** remote electronic voting systems. As stated before, all remote electronic voting processes are heavily dependent on trust due to the lack of technical understanding of the voters in the system's design. Furthermore, an attack that breaks trust is easily fabricated, does not take much effort or skill, and can thus be performed by any attacker from the attacker model. Trust is therefore fragile and easily broken in remote electronic voting processes. As mentioned before in section 6.5, trust determines whether the security requirements are met in the voter's eyes. Meeting the security requirements determines the legitimacy of a remote electronic voting process. Therefore without the voters' consensus that the security requirements are met, the results cannot be declared legitimate.

The additional downside of breaking voter trust is that it also might result in voter distrust in the remote electronic voting system itself, causing a discontinuation in usage or even doubt in future consultation results and decisions based on those results. Breaking voter trust is just as relevant for remote electronic voting systems similar in their design to **IRMA vote** as it is for all remote electronic voting systems. In conclusion, since breaking voter trust in remote electronic voting is easy, effective, and highly impactful, we consider it as the most significant risk for **all** remote electronic voting systems.

- **Biggest risk for remote electronic voting systems similar in their design to IRMA vote**

We consider the threat of changing the results, especially the attack in which a *hacker, insider, or catfisher* **targets a single voter** and deletes their vote to be the most **dangerous** threat with the **highest risk** for remote electronic voting systems similar in their design to **IRMA vote**. The reason being that most remote electronic voting systems are not end-to-end verifiable. A voter cannot make sure that, in the end, their vote is correctly counted. Remote electronic voting systems similar in their design to IRMA vote are, however, end-to-end verifiable for a voter through the published public voting register from server C. Any party can recount the votes from the public voting register and verify the results. However, the downside is that any voter can now "pretend" that their vote has been deleted from the remote electronic voting process by stating that their vote is not present in the published public voting register. As a countermeasure to this problem, IRMA implemented a so-called proof (see **section 5.3**). A voter can use this proof to prove to the party holding the voting process that they have cast a vote on server B in the voting process but that their vote is deleted from the published public voting register.

In the end, the proof and the public voting register create a vulnerability that results in attacks no longer having to be **scalable** to have a considerable impact, i.e., disrupt the voting process. Assuming that the single targeted voter verifies whether their vote was included and contributed to the final result by checking the public voting register, they will notice whether their vote was deleted. An individual voter can then use the obtained proof obtained from server B to prove that they have cast a vote on server B and consequently prove that their vote was deleted.

Deleting one vote is unacceptable since it indicates that someone has tampered with the process and its results. It also suggests a possibility of more tampering that went undetected for other cast votes. Consequently, this indicates that the *unicity* and *integrity* requirements are broken. Therefore, deleting one vote leads to a violation of the *unicity* and *integrity* requirements that results in distrust in the entire remote electronic voting process and its results, which, in the end, causes the results to be declared illegitimate. Furthermore, this distrust caused by deleting one vote might result in the other consequences earlier described in the **biggest risk for all remote electronic voting systems**.

Lastly, this vulnerability gives an obvious target to attack against which an attacker does not have to divide its resources. It is much easier to ensure that one vote is deleted from the remote electronic voting process than doing the same for many votes. The reason is that an attacker does not necessarily have to perform more complicated attacks, such as targeting the servers storing the votes. They can instead use other attacking options, such as targeting a single voter's device. In summary, this attack does not require scalability is and is possible because of the unique design of remote voting systems similar to IRMA vote.

10. Discussion

In this research, we established guiding principles, security requirements, threats, and attacker models to establish attack scenarios for remote electronic voting platforms using the **split server design** in **consultations** (see section 3.5). We looked into the details of such a system that uses the split server design, called **IRMA vote** (see chapter 5). Furthermore, we performed a risk analysis on each of these attack scenarios. Based on this risk assessment, we demonstrated that remote electronic voting platforms using the split server design are a **viable** option to use in **small-scale and low-stakes consultations**. This chapter will discuss our interpretations, implications, and limitations of these findings and give some recommendations based on these findings.

Remote electronic voting platforms using the **split server design** use two separately hosted servers to guarantee **voter secrecy** (see chapter 5). The first server checks eligibility (and therefore does know the voter's identity), while the second server only registers the cast votes (and therefore does not know the voter's identity). The second server is not required to know the voter's identity due to a **token** obtained by the voter from the first server that includes a **random blind voting number**. This token tells the second server that the voter is eligible to vote without revealing the identity of that particular voter.

Remote electronic voting platforms using the split server design are mainly based on the **secrecy** and **double voting prevention** requirements (see section 5.4). This makes remote electronic voting platforms using the split server design more attractive than digital polls used in consultations that do not adhere to these requirements (see section 3.2). Furthermore, remote electronic voting platforms using the split server design are also less expensive than postal voting while most likely increasing voter turnout, making them more attractive than postal voting (see section 3.2).

❏ 10.1. Design findings

The following section will highlight some findings that are inherent to the design of **all split server electronic voting processes**.

This research confirms generally accepted findings of preceding research in section 3.4 and section 3.5 that remote electronic voting should **not** be used in high-stakes or large-scale voting processes such as **elections**. The higher the stakes or the larger the scale of the voting process, the higher the probability of APT-level attackers and the more significant the impact of a successful cyberattack.

However, the more a consultation can be classified as **'low-stakes'** or **'small-scale**,' the less likely these problems are to occur. Furthermore, the benefits of possibly increasing voter turnout while lowering costs outweigh the possible drawbacks of cyberattacks in low-stakes and small-scale consultations. Remote electronic voting could therefore be used in low-stakes and small-scale consultations.

Moreover, current research lacks a definition of when exactly a consultation is low-stakes and small-scale. We, therefore, presented a set of characteristics that makes it possible to classify whether a consultation is low-stakes and small-scale in section 3.5 and appendix A.

To identify whether remote electronic voting systems are **secure**, we had to define what being secure means for such a system. In other words, **what** do we want to **guarantee** in the remote electronic voting system? We expressed this in the form of **security requirements** (see chapter 4) as discussed in the "Stemmen met vertrouwen" report written by the Korthals-Altes commission that advises on the Dutch electoral process ("Eindrapport Commissie-Korthals Altes 'Stemmen met vertrouwen'", 2007). From the coercion problem in remote electronic voting, it is clear that voters cannot be guaranteed to vote in liberty, and as such, this requirement from the report is, therefore, by default **broken** (see section 4.9 and section 4.10). In the end, from the eight requirements discussed in this report, six are relevant for this research.

We defined an additional security requirement, *voter turnout*, in this research (see section 4.8). This requirement was not explicitly mentioned in the "Stemmen met vertrouwen" report written by the Korthals-Altes commission ("Eindrapport Commissie-Korthals Altes 'Stemmen met vertrouwen'", 2007). We declared voter turnout as an additional requirement because a comprehensive and large voter turnout is essential for any democratic voting process, such that the decisions based on this voting process are representative and have sufficient political support.

The security requirements imply an essential **trade-off** among themselves and concerning the design of the remote electronic voting system. Designing a more *transparent* and *verifiable* remote electronic voting system naturally leads to less anonymity for the voter, i.e., less *secrecy* and vice versa. Adhering absolutely to a security requirement can lower the detection of cyberattacks or result in new attacking opportunities. Additionally, a more *secretive* voting process can make it harder to detect cyberattacks due to a lack of *verifiability* and *transparency*. On the other hand, a more *transparent* and *verifiable* process might also lead to new attacking opportunities for attackers due to the additional publicly available information that attackers can study and exploit. In the end, there needs to be a balance between all security requirements.

Lastly, the coercion problem also implies that the *secrecy* requirement is partially broken in remote electronic voting. It is partially broken, i.e., **weak**, since the split server design provides the voter anonymity as long as they do not vote while other people are present. Furthermore, it is worth mentioning that because the *security* and liberty requirements are infringed and broken in remote electronic voting processes, remote electronic voting platforms are **unsuitable** for any Dutch **election** since the eight requirements must be upheld by law in (official) Dutch elections.

Furthermore, we established relevant **threats** in this research to determine how **secure** remote electronic voting platforms using the split server design are (see chapter 6). The first four threats represent a more specific notion of the regularly used confidentiality, integrity, and availability requirements, while the fifth threat of **breaking voting trust** did not (see section 6.5). Mainly, preventing the threat of **undetected** attacks that change the voting results is desirable for remote electronic voting systems, i.e., attacks on integrity, since undetected attacks have the most significant impact and are the most undesirable.

This research additionally established six **attacker models** to determine the capabilities and goals of the relevant attackers that will violate security requirements (see chapter 7). We consider the *insider threat* (see section 7.1) as the most **dangerous** attacker due to insider threats possessing profound knowledge of the consultations remote electronic voting system and possessing access to vital parts of the system making it more likely that they circumvent the split server design and conduct an undetected attack. As a consequence, when these insiders pose a threat, the impact will be more significant. Furthermore, we consider the *influencer* (see section 7.5) and the *mother with the baseball bat* (see section 7.2) as the most **likely** attackers since both do not require much effort or skill and are quickly effective.

❏ 10.2. Implementation findings

The following section will highlight some findings inherent to the implementation of split server electronic voting processes in remote electronic voting platforms similar in their design to **IRMA vote**.

The security requirements, threats, and attacker models together form the **attack scenarios** in which an attacker violates a security requirement (realizes a threat) and makes the system **insecure** (see chapter 8). **Attack 24** (see section 9.4) is the most noteworthy attack due to an attacker being able to target and delete the vote of a single voter and still have a scalable impact on the remote electronic voting process. The reason being is that voters could prove that their vote has been deleted from the voting process. Voters could do so by verifying whether their vote is present in the published public voting register and by having the proof from server B that demonstrates that they have cast a vote. Furthermore, **attack 5** (see section 9.4) is also noteworthy since it is only made possible due to municipalities not wanting two separate voting phases (see section 5.2).

In the end, the **risk analysis** of each attack scenario determines whether remote electronic voting systems using the split server design are **secure** (see chapter 9)**. Our findings suggest that all attack scenarios either pose a medium or a high risk.** The most **significant risk** present in **all** remote electronic voting systems is breaking voter trust by any attacker from the attacker model (see section 9.6). Breaking voter trust poses the most significant risk because it quickly leads to voter distrust in the results and future usage of remote electronic voting systems while requiring little skill or effort. It is especially easy to break voter trust since it is more fragile in nature when using remote electronic voting. The **most significant risk** for remote electronic voting systems **using the split server design** is changing the results. Primarily when a *hacker, insider, or catfisher* targets a single voter and deletes their vote due to the reasons mentioned earlier.

We consider the amount of verifiability and transparency that remote electronic voting systems similar to IRMA vote offer as a **mitigation** of the impact regarding attacks that break voter trust. We consider that the benefits of the public voting register **outweigh** the possibility of an attacker targeting a single voter exploiting this feature. The benefits being that there is a verifiability possibility that mitigates the likelihood and impact of various attacks. The public voting register ensures that many attacks become easily detectable, which mitigates the impact of attacks that commit voter fraud or change the results of the voting process. If attacks become detected, the voting process can be held again, which we consider an acceptable mitigation of an attack. We, therefore, conclude that the two most significant risks are **manageable**.

❏ 10.3. Limitations

One limitation of this research concerns the assumption that voters check whether their vote is present and correct in the public published voting register. If voters do not verify their vote in the public voting register, many cyberattacks can go unnoticed, and thus the likelihood of an attack succeeding increases.

Vote-selling was deemed out of this research scope since it is impossible to stop vote-selling in remote electronic voting processes due to the uncontrolled environment (see chapter 7). Furthermore, the publicly published voting record of remote electronic voting systems similar in their design to IRMA vote may allow attackers to offer a reward to anyone who can prove that they have voted in a particular way, making vote-selling even more likely (Botros et al., 2021). Nevertheless, the question remains how likely and significant these attacks are in low-stakes and small-scale remote electronic voting processes and therefore poses another limitation to this research.

In this research, we left out attacks based on breaking the cryptography of remote electronic voting systems and instead referred to Doesburg's (2020) bachelor thesis. However, since all cybersecurity is essentially based on cryptography, this poses an apparent limitation to our research.

Lastly, another limitation in this research involves forgotten attacks that could pose a high risk. The attacks described in the attack scenarios are not exclusive, and each attack scenario might consist of other overlooked attacks.

❏ 10.4. Recommendations

Remote electronic voting systems using the split server design adhere strongly to the security requirements and fend off established attack scenarios based on threats and attacker models in this research well. Therefore, remote electronic voting systems using the split server design are **viable** options for **small-scale** and **low-stakes** consultations. The reason is that the more a consultation can be classified as low-stakes or small-scale, the more the risk of each cyberattack scenario is **limited** by the split server design of the remote electronic voting platform, while other risks are adequately **manageable** from an organizational point of view. Nevertheless, the exact line between 'low-stakes' and 'high-stakes' and 'small-scale' and 'large-scale' should remain a **topic of debate** and should be **uniquely** defined for each consultation context based on what the organizer of the consultation considers as acceptable risks since it is an essential part of the rationale to recommend this remote electronic voting format.

11. Future Work

This chapter will describe problematic or untouched areas regarding split server electronic voting processes that could still be researched in the future.

- For starters, this research and commonly accepted preceding research have repeatedly indicated the predicament that trust brings to remote electronic voting. This research has not focused on the effect the user experience of a remote electronic voting system might have on voters' trust in the cybersecurity and integrity of the remote electronic voting process. This might pose an interesting research topic that requires further research.

- Voters tend to lack technical knowledge in topics such as cryptography, on which the cybersecurity of the remote electronic voting process depends. Therefore, it might be interesting to research whether voters start trusting remote electronic voting platforms, such as IRMA, to a higher or to smaller degree after watching a small educational video. This might indicate whether education helps or not in battling this problem.

- This research does not go in-depth into attacks that break the cryptography of remote electronic voting systems similar in their design to IRMA vote. Doesburg (2020) already made an in-depth analysis of the cryptography of IRMA. However, further research that performs a similar risk analysis as conducted in this research, on attacks purely focused on breaking the cryptography, might prove valuable. Alternatively, research that performs penetration testing on the system might also be beneficial.

- A noticeable vulnerability specific to IRMA-based remote electronic voting is that servers A and B are hosted on behalf of the municipality (see section 5.4). An interesting topic for further research might be creating a design that helps mitigate this vulnerability's risk.

- This research does not go in-depth into the organizational procedures required for a secure remote electronic voting system. Further research might be conducted on the topic of organizational design that results in a viable and secure remote electronic voting process.

- We decided not to research attacks concerning vote-selling and vote-buying. However, the question remains how likely and significant these attacks are in low-stakes and small-scale remote electronic voting processes. This might be another interesting topic for future research.

- Lastly, remote electronic voting systems similar in their design to IRMA vote do not, at the moment, possess the functionality of being able to revoke and change votes due to, among others, cybersecurity concerns. Designing, developing, and securely implementing these functionalities might be another interesting topic of research.

12. Conclusion

In this research, **cyberattack scenarios** were developed based on security requirements, threats, and attacker models for **consultations** using remote electronic voting platforms based on a **split server electronic voting process**. In such a process, one server checks eligibility (and therefore does know the voter's identity), while another server only registers the cast votes (and therefore does not know the voter's identity). This split server design guarantees voter **secrecy**. We looked into the details of such a remote electronic voting system that uses the split server design, called **IRMA vote**. In addition, we conducted a **risk assessment** for each attack scenario.

This research indicates that using remote electronic voting in 'high-stake' and 'large-scale' voting processes, such as elections, is **unwise**, as is generally accepted in preceding research and discussed in section 3.4 and section 3.5. It is unwise since there is an increased probability of attacks and higher-level attackers and since a successful attack has a more considerable impact. In 'small-scale' and 'low-stake' voting processes, these problems are less likely to occur, and the impact tends to be less substantial. In addition, remote electronic voting in small-scale and low-stakes scenarios strives to increase voter turnout (accessibility) and lower costs. When realized, these remote electronic voting benefits outweigh the drawbacks in low-stake and small-scale scenarios. Remote electronic voting in **low-stake and small-scale** scenarios, such as consultations, is therefore an **attractive** option.

Furthermore, this research has indicated three reasons that split server electronic voting processes are an attractive alternative to currently used voting methods in low-stake and small-scale consultations. Firstly, split server electronic voting processes adhere as much as possible to the **security requirements** (chapter 4) by design (chapter 5). Secondly, split server electronic voting processes are more **affordable** than using postal voting for consultations. They also have measures in place to **prevent double-voting**, which the currently used digital polls in consultations do not have (as discussed in section 3.2). Finally, split server electronic voting processes indicate that they can fend off probable attack scenarios (chapter 8) based on the established security requirements (chapter 4), threats (chapter 6), and attacker models (chapter 7) of this research with **acceptable risks** (chapter 9). Split server electronic voting processes can fend off probable attack scenarios with acceptable risks (chapter 9), mainly due to the split server design, the obtained token by the voter (random blind voting number), the published public voting register, and the QR-codes used in the voting process. Moreover, other risks are adequately **manageable** from an organizational point of view.

In conclusion, split server electronic voting processes can function as a **viable** approach to support remote electronic voting processes in low-stake and small-scale consultations with manageable risks.

13. References

Aarts, C. W. A. M. (1999). *Opkomst bij verkiezingen: Onderzoeksrapportage in opdracht van het Ministerie van Binnenlandse Zaken en Koninkrijksrelaties, Directie Constitutionele Zaken en Wetgeving. (Bijlage bij: Tweede Kamer, 1998-1999, 26200, vii, nr.61).* Ministerie van Binnenlandse Zaken/Justitie.

Ajish, S & AnilKumar, K. (2020). Secure I-voting system using QR code and biometric authentication. *Information Security Journal: A Global Perspective*, 1-22. https://doi.org/10.1080/19393555.2020.1867261

Augoye, V. and Tomlinson, A. (2018). Analysis of electronic voting schemes in the real world. In: *UK Academy for Information Systems Conference Proceedings 2018.*

Buchsbaum, T.M. (2004). E-Voting: International Developments and Lessons Learnt. *Electronic Voting in Europe Technology, Law, Politics and Society*, 31-42.

Botros, L., van Gastel, B., Jacobs, B., & Schraffenberger, H. (2021). Attribute-based E-Voting for small scale Elections [Draft version, still under peer review].

Chen, P., Desmet, L., & Huygens, C. (2014). A Study on Advanced Persistent Threats. In: *De Decker B., Zúquete A. (eds) Communications and Multimedia Security. CMS 2014. Lecture Notes in Computer Science*, *vol 8735*. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-44885-4_5

Doesburg, J. (2020). *Using IRMA for small scale digital elections* [Bachelor Thesis, Radboud University]. Cs.ru.nl. Retrieved 25 January 2021, from https://www.cs.ru.nl/bachelors-theses/2020/Job_Doesburg___4809327___Using_IRMA_for_small_scale_digital_elections.pdf

Ehringfeld, A., Naber, L., Kappel, K., Fischer, G., Pichl, E., & Grechenig, T. (2011). Learning from a Distributed Denial of Service Attack against a Legally Binding Electronic Election: Scenario, Operational Experience, Legal Consequences. In: *Andersen K.N., Francesconi E., Grönlund Å., van Engers T.M. (eds) Electronic Government and the Information Systems Perspective*. *EGOVIS 2011. Lecture Notes in Computer Science*, *vol 6866.* Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-22961-9_5

Eindrapport Commissie-Korthals Altes 'Stemmen met vertrouwen'. Kiesraad.nl. (2007). Retrieved 24 January 2021, from https://www.kiesraad.nl/adviezen-en-publicaties/rapporten/2007/09/27/eindrapport-commissie-korthals-altes-stemmen-met-vertrouwen

Eindrapport Commissie-Van Beek 'Elke stem telt'. Kiesraad.nl. (2013). Retrieved 15 February 2021, from https://www.kiesraad.nl/adviezen-en-publicaties/rapporten/2013/12/18/eindrapport-commissie-van-beek-elke-stem-telt

van Gastel, B. (2021). *IRMA meeting 5 Maart, Locaal stemmen met IRMA* [Slides from presentation]. Privacy by Design Foundation. Retrieved 14 March 2021, from https://privacybydesign.foundation/meetings/

Gibson, R. (2001). Elections Online: Assessing Internet Voting in Light of the Arizona Democratic Primary. *Political Science Quarterly*, *116*(4), 561-583. https://doi.org/10.2307/798221

Gritzalis, D. (2002). Principles and requirements for a secure e-voting system. *Computers & Security*, *21*(6), 539-556. Amsterdam: Elsevier. https://doi.org/10.1016/s0167-4048(02)01014-3

Heiberg, S., & Willemson, J. (2014). Verifiable internet voting in Estonia. *IEEE 2014 6Th International Conference On Electronic Voting: Verifying The Vote (EVOTE)*. https://doi.org/10.1109/evote.2014.7001135

Irma-demo.stemmen.stempas - IRMA credentials. (2021). Privacy By Design foundation [IRMA documentation]. Retrieved 25 February 2021, from https://privacybydesign.foundation/attribute-index/en/irma-demo.stemmen.stempas.html

IRMA-vote. Interdisciplinary Hub for Security, Privacy and Data Governance. (2021). [IRMA documentation]. Retrieved 25 February 2021, from https://www.ru.nl/ihub/research/research-projects/irma-vote

ISO/IEC 27000. (2018). International Standard ISO / IEC Information technology — Security techniques — Information security management systems — Overview and. *ACM Workshop on Formal Methods in Security Engineering.Washington, DC, USA*, *34*(19), 45–55.

Jacobs, B., & Pieters, W. (2009). Electronic Voting in the Netherlands: From Early Adoption to Early Abolishment. In: *Aldini A., Barthe G., Gorrieri R. (eds) Foundations of Security Analysis and Design V. FOSAD 2009, FOSAD 2007, FOSAD 2008. Lecture Notes in Computer Science*, *vol 5705*. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-03829-7_4

Jefferson, D., & Rubin, A., & Simons, B., & Wagner, D., (2004). *A Security Analysis of the Secure Electronic Registration and Voting Experiment (SERVE).* http://www.serveusa.gov/

Jonker, H., Mauw, S., & Pang, J. (2013). Privacy and verifiability in voting systems: Methods, developments and trends. *Computer Science Review*, *10*, 1-30. Amsterdam: Elsevier. https://doi.org/10.1016/j.cosrev.2013.08.002

Krimmer, R., Volkamer, M., & Duenas-Cid, D. (2019, October). E-voting–an overview of the development in the past 15 years and current discussions. *In the International Joint Conference on Electronic Voting*, 1-13. Springer, Cham.

Krimmer, R., Triessnig, S., & Volkamer, M. (2007). The Development of Remote E-Voting Around the World: A Review of Roads and Directions. In: *Alkassar A., Volkamer M. (eds) E-Voting and Identity. Vote-ID 2007. Lecture Notes in Computer Science, vol 4896*. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-77493-8_1

Kulyk, O., & Neumann, S. (2020). Human Factors in Coercion Resistant Internet Voting–A Review of Existing Solutions and Open Challenges. In *Proceedings of the Fifth International Joint Conference on Electronic Voting.* TalTech press.

Lijphart, A. (1998). The Problem of Low and Unequal Voter Turnout - and What We Can Do About It. *IHS Political Science Series* No. 54, February 1998.

Maaten, E. & Hall, T., (2008). Improving the Transparency of Remote E-Voting: The Estonian Experience. In: *Krimmer, R. & Grimm, R. (Hrsg.), Electronic Voting 2008 (EVOTE08). 3rd International Conference on Electronic Voting 2008, Co-organized by Council of Europe, Gesellschaft für Informatik and EVoting.CC.* Bonn: Gesellschaft für Informatik e. V., 31-43.

Martin, J. (1983). *Managing the data-base environment. Prentice-Hall* PTR.

Mote, C. D. (2000). Report of the national workshop on internet voting: issues and research agenda. In *Proceedings of the 2000 annual national conference on Digital government research (dg.o '00)*, 1–59 Digital Government Society of North America.

Municipality of Groningen. (2021). *IRMA Voting flow diagram* [Image].

Okediran, O., Olabiyisi, S., Omidiora, E., & Ganiyu, R. (2011). A Survey of Remote Internet Voting Vulnerabilities. *World Of Computer Science And Information Technology Journal (WCSIT), Vol. 1*, No. 7, 297-301.

Ostaaijen, J., Epskamp, M., & Dols, M. (2016). *Verbetering op komst: Een verkenning naar een effectieve gemeentelijke inzet van communicatiemiddelen voor de opkomst bij lokale verkiezingen.* Tilburg University.

Petitpas, A., Jaquet, J., & Sciarini, P. (2020). Does E-Voting matter for turnout, and to whom?. *Electoral Studies*, 102245. Amsterdam: Elsevier. https://doi.org/10.1016/j.electstud.2020.102245

Pieters, W. (2008). *La Volonté Machinale: Understanding the Electronic Voting Controversy* [Doctoral dissertation, Radboud University]. Repository.ubn.ru.nl. Retrieved 16 February 2021, from https://repository.ubn.ru.nl/handle/2066/32048;jsessionid=DFB38AFE4BAA0E47D03933D62FAFE1A5

Privacy by Design Foundation. (2021). Privacy by Design Foundation [IRMA documentation]. Retrieved 24 January 2021, from https://privacybydesign.foundation/irma-explanation/

Raadplegingen met IRMA. (2021). Election-register.sustainablesoftware.info [IRMA documentation]. Retrieved 26 February 2021, from https://election-register.sustainablesoftware.info/

Randomblind issuance IRMA docs. (2021). Irma.app [IRMA documentation]. Retrieved 23 March 2021, from https://irma.app/docs/randomblind/

Saxena, N., Hayes, E., Bertino, E., Ojo, P., Choo, K.-K. R., & Burnap, P. (2020). Impact and Key Challenges of Insider Threats on Organizations and Critical Businesses. *Electronics*, *9*, 1460. https://doi.org/10.3390/electronics9091460

Schryen, G., & Rich, E. (2009). Security in Large-Scale Internet Elections: A Retrospective Analysis of Elections in Estonia, The Netherlands, and Switzerland. *IEEE Transactions On Information Forensics And Security*, *4*(4), 729-744. https://doi.org/10.1109/tifs.2009.2033230

Stemmen tellen met de hand. Tweedekamer.nl. (2017). Retrieved 3 March 2021, from https://www.tweedekamer.nl/zo-werkt-de-kamer/verkiezingen-en-kabinetsformatie/verkiezingen-2017/stemmen-tellen-met-de-hand

Solvak, M., & Vassil, K. (2016). *E-voting in Estonia: Technological Diffusion and Other Developments Over Ten Years (2005 - 2015)*. University of Tartu.

Springall, D., Finkenauer, T., Durumeric, Z., Kitcat, J., Hursti, H., MacAlpine, M., & Halderman, J. (2014). Security Analysis of the Estonian Internet Voting System. *Proceedings Of The 2014 ACM SIGSAC Conference On Computer And Communications Security*, 703–715. https://doi.org/10.1145/2660267.2660315

Technical overview IRMA docs. (2021a). Irma.app [IRMA documentation]. Retrieved 23 March 2021, from https://irma.app/docs/overview/#attribute-based-signatures

Technical overview IRMA docs. (2021b). Irma.app [IRMA documentation]. Retrieved June 11, 2021, from https://irma.app/docs/overview/#irma-security-properties

Trechsel, A., Kucherenko, V., & Silva, F. (2016). *Potential and challenges of e-voting in the European Union.* European Parliament's Committee On Constitutional Affairs, Policy Department For Citizens' Rights And Constitutional Affairs.

Verbij, R. (2014). *Risk assessment framework based on attacker resources Services, Cyber Security and Safety group*. [Doctoral dissertation, Radboud University]. Essay.utwente.nl. Retrieved 27 January 2021, from http://essay.utwente.nl/65811/1/Verbij_MA_EMCS.pdf

What is IRMA? IRMA docs. (2021). Irma.app. [IRMA documentation]. Retrieved 25 February 2021, from https://irma.app/docs/what-is-irma/

Willemson, J. (2018). Bits or paper: Which should get to carry your vote? *Journal Of Information Security And Applications*, *38*, 124-131. Amsterdam: Elsevier. https://doi.org/10.1016/j.jisa.2017.11.007

Zissis, D., & Lekkas, D. (2011). Securing e-Government and e-Voting with an open cloud computing architecture. *Government Information Quarterly*, *28*(2), 239-251. Amsterdam: Elsevier. https://doi.org/10.1016/j.giq.2010.05.010

14. Appendix

❑ A. Decision tree

If one of the answers to the decision tree factors' questions is 'yes,' it is a high stake or large scale voting process, and remote electronic voting, such as IRMA vote, should not be used. If all the answers to all the factors are 'no,' the stakes and scale are in theory low and small, and remote electronic voting, such as IRMA vote, can be used. Nevertheless, for each case, the decision tree is only a guideline and should not be blindly followed. In the end, it is essential to realize that the exact line between 'low-stakes' and 'high-stakes' and 'small-scale' and 'large-scale' should remain a topic of debate and should be uniquely defined for each consultation context.

• Tree factors

■ Scale

High scale: The total number of eligible voters is greater than 10.000, or the outcome affects more than 20% of a municipality's population.

■ Societal influence

High stakes: The voting process concerns one of the following:

- Judicial laws;
- Elections of a human;
- Wages (both individual and group);
- Extensive business activities with a revenue greater than 1.000.000 euros;
- Healthcare;
- Human and animal rights;
- Public utilities, like energy, water, sanitation, etc;
- Significant environmental or hazard risk and impact.

■ Reversibility

High stakes: The outcome cannot be reversed within less than a month.

■ Cross Border Effects

High stakes: The outcome's effect does not reach beyond its intended region. It does not affect the following areas in any way:

- Other countries;
- Other provinces;
- Other municipalities;
- Or has national effects.

Amount of money

High stakes: The amount of money involved is greater than one million euros.

Individual effect

High stakes: The result does distribute political, enforcement, or emergency service powers to an individual or a group.

Decision tree

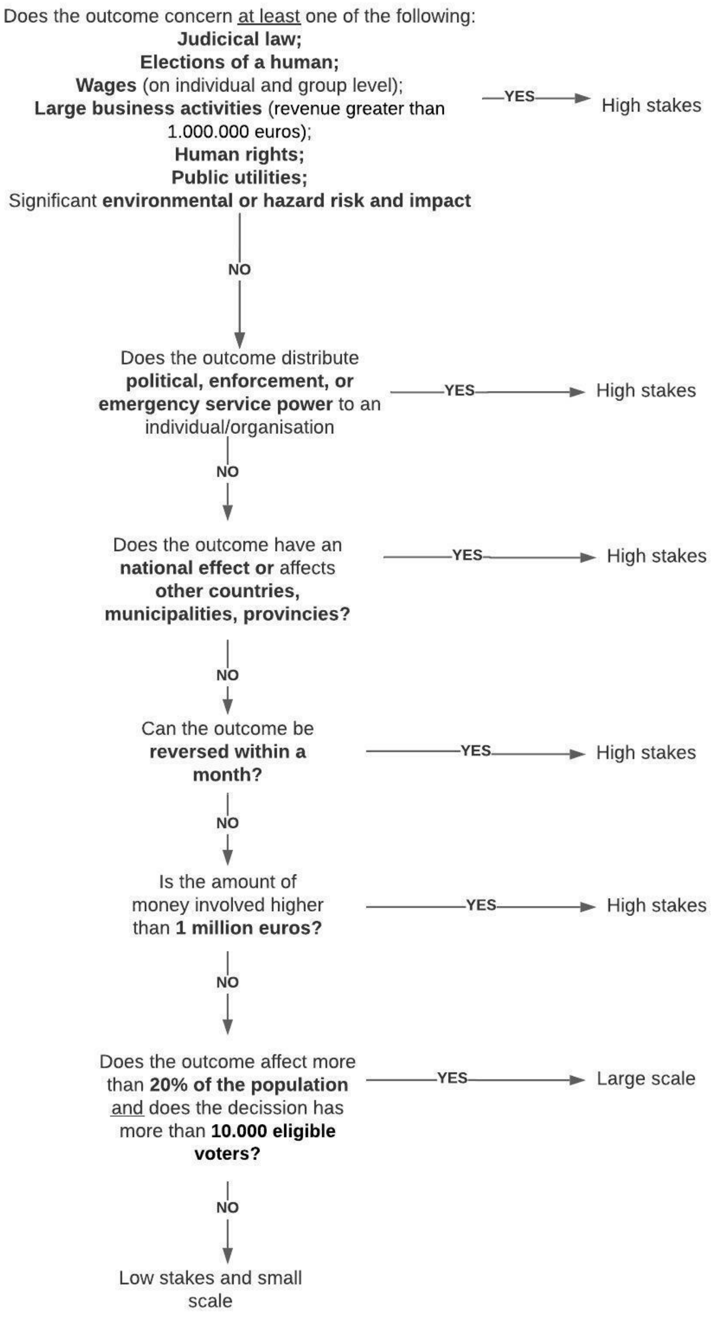

Does the outcome concern at least one of the following:
Judicical law;
Elections of a human;
Wages (on individual and group level);
Large business activities (revenue greater than 1.000.000 euros);
Human rights;
Public utilities;
Significant environmental or hazard risk and impact
YES
High stakes
NO
Does the outcome distribute political, enforcement, or emergency service power to an individual/organisation
YES
High stakes
NO
Does the outcome have an national effect or affects other countries, municipalities, provincies?
YES
High stakes
NO
Can the outcome be reversed within a month?
YES
High stakes
NO
Is the amount of money involved higher than 1 million euros?
YES
High stakes
NO
Does the outcome affect more than 20% of the population and does the decission has more than 10.000 eligible voters?
YES
Large scale
NO
Low stakes and small scale

Figure 3. Depicts a decision tree to determine whether a consultation can be classified as 'low-stakes or 'high-stakes' and 'small-scale' or 'large-scale.'

B. Overview for IRMA-based remote electronic voting

(Botros et al., 2021)

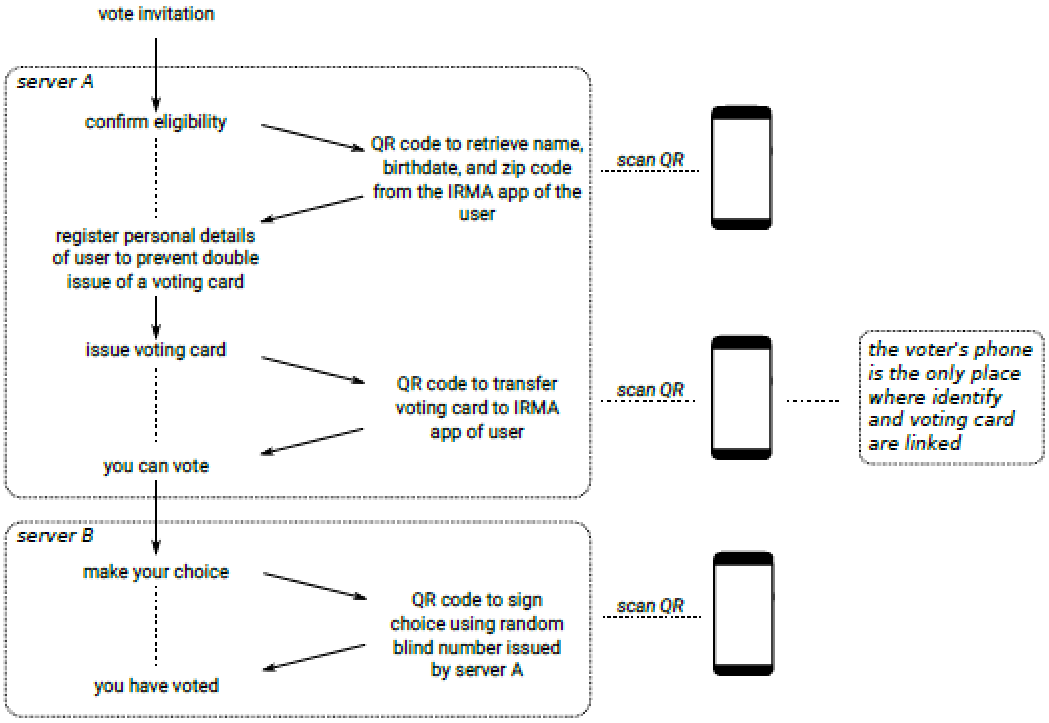

vote invitation
server A
confirm eligibility
QR code to retrieve name, birthdate, and zip code from the IRMA app of the user
scan QR
register personal details of user to prevent double issue of a voting card
issue voting card
QR code to transfer voting card to IRMA app of user
scan QR
the voter's phone is the only place where identify and voting card are linked
you can vote
server B
make your choice
QR code to sign choice using random blind number issued by server A
scan QR
you have voted

Figure 4. Depicts an overview of the interaction between the two different servers (websites), server A and server B.

❑ C. Flow diagram for IRMA vote

(Municipality of Groningen, 2021)

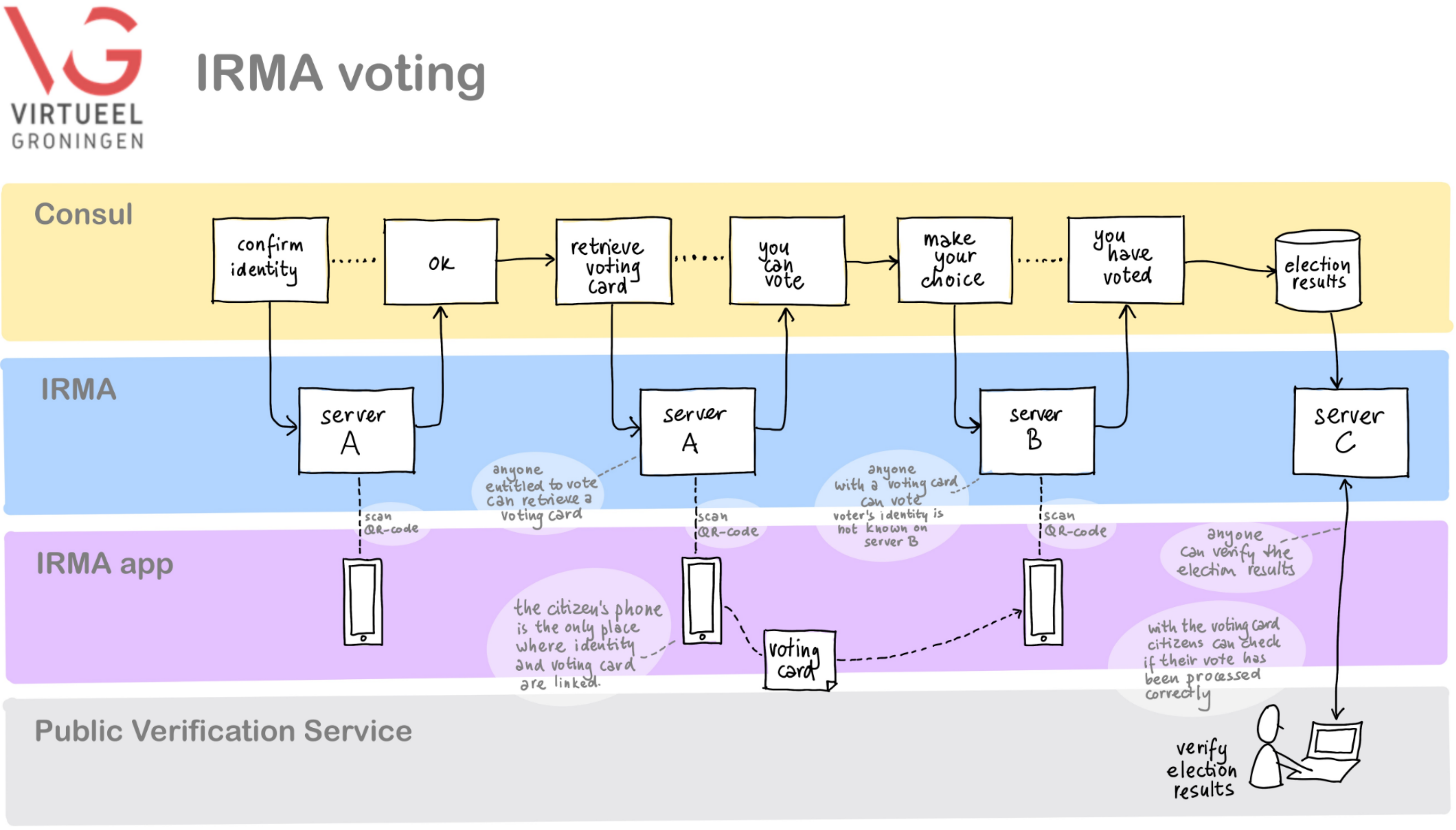

VIRTUEEL GRONINGEN
IRMA voting
Consul
IRMA
IRMA app
Public Verification Service
confirm identity
ok
retrieve voting card
you can vote
make your choice
you have voted
election results
server A
server A
server B
server C
scan QR-code
scan QR-code
scan QR-code
anyone entitled to vote can retrieve a voting card
anyone with a voting card can vote
voter's identity is not known on server B
anyone can verify the election results
the citizen's phone is the only place where identity and voting card are linked.
voting card
with the voting card citizens can check if their vote has been processed correctly
verify election results

Figure 5. Depicts a flow diagram of the interaction between the three different servers (websites), A, B, and C.