# Project Report: An analysis of the 2016 U.S. election cyberattacks and impacts of technical hacking vs social engineering techniques[*]

Anonymous Author(s)

## ABSTRACT

The 2016 U.S. election was the target of an unprecedented Russian misinformation campaign. Although, more traditional cyberattacks were performed against the U.S. and its organizations, the leading attack vector was through the use of online trolls. With the use of social media and online political campaigns, it has become easier for Russia, or any other adversary, to interfere and spread misinformation by targeting people, rather than organizations or U.S. agencies. The recent 2016 election, thus, saw a rise in the use of social engineering attacks, as opposed to more technical hacking-based attacks. We review the difference and impact social engineering attacks can have vs. more traditional cyberattacks in the context of elections and public opinion, and apply these observations to creating a secure voting platform. We take into consideration the cybersecurity knowledge areas and discuss how we apply them to an online application that can be utilized by governments to hold secure elections. This was achieved by applying modern best-practices and techniques, such as SSL encryption, hashed passwords, two-factor authentication, and running multiple services in different, isolated virtual machines.

## KEYWORDS

cybersecurity, cyberattacks, social engineering, social media, elections, public opinion, security

## 1 INTRODUCTION

The Soviet Union and now Russia under Vladimir Putin have waged a political power struggle against the West for nearly a century. Now, through the use of spreading false and distorted information. These misinformation strategies attempt to interrupt the ability to have a level-headed political discourse. This is only furthered as around 14% of Americans (in 2016) said that their primary method of getting their news through social media, and people are also more likely to believe stories that favor their preferred candidate, especially if they have ideologically segregated social media networks [1].

The primary focus of this research will to compare the effectiveness, impact, and scalability of technical hacking cyberattacks versus social engineering attacks. With the rise of social media, it has never been easier to communicate with people and share information. It has also been a crucial advertisement space for small and large organizations, which also opens the possibility for abuse. Increasingly, political actors are automating the spread of political communication through algorithms that obscure motives and authors yet reach immense networks of people[6]. When large organisations, or even governments, get involved in the misinformation campaign, they bring in resources that allow them do so with ease. The lack of a need for a specialized skill set to become an agent of misinformation allows an organization to easily scale the task to allow for more widespread damage, this is what potentially makes the attack so lethal: skewing public opinion to alter the outcome of a powerful appointed official.

Additionally, we will dive into current methods/frameworks being researched and deployed by researchers and experts in the field today to detect and prevent cyberattacks during es littlethe 2020 election and any other future elections. This includes organizations such as Facebook and Google, to name a few, who are actively working law enforcement and intelligence officials to discuss how to better align efforts to defend the 2020 election. The main approaches taken by the organizations is to continue to build a stronger partnership that allows for greater transparency[4].

### 1.1 Attack Vectors

#### 1.1.1 *Social Engineering.*

These strategies are also easily employed, as it does not require a skilled hacker to spread misinformation, which led Russia to build companies whose sole purpose was to hire people to spread political misinformation. Employees would pose as Americans and would create racially/politically divisive social media groups/pages, in addition with fake news articles and commentaries to build political animosity within the American public. Additionally, the Russian military intelligence agency, know as GRU, allegedly used coordinated hacking to target people and institutions in the U.S.. Russian hackers were, therefore, able to leak damaging information through multiple aliases, a couple of which are, but not limited to, “DCLeaks” and “Guccifer 2.0.”

#### 1.1.2 *Phishing campaign.*

Technically a subset of social engineering, phishing is when people are sent an email or message that looks legitimate or offers some sort of reward. The ultimate goal is to get the individual to click on malicious link that can then install malicious software, collect credentials or be used for any one of a large number of nefarious purposes. These kinds of attacks are very prevalent nowadays, especially within corporations and organizations.

Under the umbrella term of phishing, there are a 5 most common types. Of these, the most common is email phishing. Email phishing is when attackers send out a blast email, playing the odds that someone will click on the malicious link. Another big type is spear-phishing which are sent to specific people or a specific group of people. The third type is a variation of spear-phishing called Whaling. This is a hyper-focused attack on senior executives and are

[*] Title note

generally a different form than what's used for other types of phishing. There are 2 other slightly more minor types that are growing in popularity. The first is smishing and vishing. Smishing and vishing are phishing through text messages and phone calls. Last but not least, there's angler phishing. Angler phishing is phishing through social media.[7]

Some of the most prominent examples of phishing are the attacks on the DNC, leading up to the 2016 presidential election. Specifically, these attacks are an example of spear-phishing. Attackers sent out 2 rounds of emails. The first sent an email to 1000 email addresses, where they were tricked into opening a malicious file. This kicked off a chain of events that led to emails being taken from several DNC accounts and exfiltrated to the attackers infrastructure. Hillary Clinton's emails ended up being a major scandal during the election, possibly swaying public opinion in favor of Trump. There was also a second attack that led to people's passwords being taken. There's less information on this second attack but it's likely that information was taken from several senior members in the party. [3]

#### 1.1.3 Technical hacking.

Foreign attacks on US elections weren't limited to just social attacks. Evidence points to infiltration of electronic voting systems by Russian agents during the 2016 election. This all starts in Durham County, NC. In order to set up the election system, county officials would load voter data onto 2227 USB flash drives and then insert those flash drives into laptops. It was observed that this process took 8-10 times longer than usual. This unusually long time was starting to jeopardize preparations for the election. In response to this, they contacted their contractor, VR Systems, who remotely connected to their desktop in order to try and diagnose the issue. This is a pretty common practice for IT services when trying to solve a problem. However, this also obviously opens up the individuals computer to attack. It appears this is what occurred to Durham County, because almost immediately after being remotely accessed, the laptops started crashing, freezing, refusing to let people vote and a host of other issues. This in turn force Durham County to abandon electronic voting and use paper ballots instead.

As it turns out, 3 months earlier in the election, VR Systems had been targeted by Russian hackers in a phishing campaign. The hackers had sent malicious emails to VR Systems and some of their customers in an attempt to get the username and password for their email accounts. Russian hackers also visited VR Systems' website in an effort to find vulnerabilities like they did with Illinois' voter registration. VR Systems had a third-party company conduct a forensic report on the company which ended up not finding evidence Russian's had compromised their system. However, this happened a year after the attack allegedly happened, which leaves more then enough time for the Russians to remove any evidence of having been on the system. Worryingly, it's theoretically possible for a hacker to get into these e-voting systems and actually change the number of votes in order to swing the election. This actually happened where one of the systems began showing election results 30 minutes before voting ended, which is a violation of state law. The hackers actually sent emails from an address appearing to be from VR Systems with an updated user guide. If downloaded, this word document would open up a backdoor on the system to the hackers. It appears 2 Florida counties and 1 unknown county clicked on these emails. VR Systems didn't know about these attacks until November 1st, just 7 days before the election. VR Systems sent out an email regrading these campaigns, but didn't mention the attacks that had happened to VR Systems' themselves in August.

It took almost a year, June 2017, for the public to learn about the possible hacking of VR Systems. Only referred to as U.S. Company 1, the description was so close, VR Systems even though it was them and reached out to the FBI to ask why the NSA was saying they'd been hacked. Come to find out in the 2018 indictment of 12 Russian Military officers, it's said they hacked in the computers of US Voting software vendors, and sent over 100 phishing emails to people administering elections in Florida. The Mueller report also says that the hackers managed to install malware on the network of an unknown company, though the report also notes that an effort wasn't made to verify these findings.

### 1.2 Current Progress Towards Social Engineering Mitigation Strategies

The current administration has not made impactful progress towards the mitigation of future electoral interference. Although, President Donald Trump convened a pro forma National Security Council meeting to address the topic, no major new authorities or initiatives resulted. He subsequently also signed an executive order to punish perpetrators of election interference, but its effectiveness remains unclear[2].

According to the Securing American Elections report[8], it should be expected that additional domestic and foreign actors will join Russia in future attempts to use social media and cyber technologies to interfere with U.S. elections. The report is compromised of eight chapters that identify ongoing issues and recommendations:

(1) Increasing the security of the U.S. election infrastructure
(2) Enhancing transparency about foreign involvement in U.S. elections
(3) Confronting efforts at election manipulation from foreign media organizations
(4) Combating organized disinformation campaigns from state-aligned actors
(5) Regulating online political advertising by foreign governments and nationals
(6) Establishing international norms and agreements to prevent election interference
(7) Deterring foreign governments from election interference

The report concludes with the authors stating that U.S. elections must be free and fair, Americans must choose their leaders alone, free from interference.

#### 1.2.1 Social Media Policy Framework.

In this section, we will be exploring social media framework that protects companies and analyze how the framework can be applied in election systems. There has been increasing number of people that has been using social media within professional settings that has caused companies multitude of security issues. These issues affect company's employee, process and technology. Social engineers utilizes the information that they have collected to victimize these companies. By utilizing ICT security policy control, companies can reduce the risk of becoming a victim of social engineering [9]. This framework as been developed by observing the challenges that the social engineers needs to go through to get the information that they might need. By analyzing the tactics that social engineers use alongside modern security practices to create Social Engineering through Social Media (SESM) framework by analyzing company's system implementation and incorporating appropriate security standards within the system. By doing so, company's system will be less likely to affected by social engineering attacks. Similarly, by conceptualizing election system and implementing appropriate security standards will reduce social engineering attacks.

# 2 OUR APPROACH

## 2.1 Motivation

Online voting systems are notoriously unsafe and risky. There are countless stories of online voting systems or machines breaking, crashing, or getting hacked[5]. This can be attributes to a number of reasons, some of which include (but are not limited to) DDOS attacks, unsecure connections, un-encrypted databases, outdated software/OS. We decided to tackle the implementation of such system to address many of these issues. As voting systems and everyday applications slowly transform to online services, they are at a greater risk of cyberattacks, but they do pose a great advantage for absentee ballots or vote-by-mail replacements. That is why such a tool could potentially be a crucial aspect that allows more people to vote. We continue to read about how countless companies have failed to produce a robust voting system, to motivate us as well as learn from their systems' pitfalls.

# 3 PROJECT GOALS

We chose to develop a secure API that allows any user-facing application to seamlessly and safely connect to allow a person to cast an immutable vote. This was done by building the voting application's backend using the Go language and connecting it to a MySQL database. Both systems are deployed using Docker containers, which allows the application to be deployed from virtually any environment or cloud service and be scaled easily to avoid heavy congestion. We utilize the principle of least privilege and fail-safe default, as well as applying roles to users upon account creation. We have opted for as simple of a design to mitigate undesired "features" and behaviours.

# 4 PROJECT DESIGN

## 4.1 User Design and Experience

The current application design implements a system that allows users to sign up and vote for their favorite candidate, and allows candidates to sign up and view their total counts. A user is then able to sign up, choose a party which they wish to be affiliated with, create a username with password, and log in to cast their vote for their preferred candidate. A candidate can then log in and view votes cast anonymously, or view the number but not the users who cast them. To increase the security of the application, two factor authentication (2FA) was added. For the project, users must go through Google Authenticator to authenticate themselves. When the user first creates an account, user will be instructed to take an image of QR code to link the user with the phone. The authenticator app will generate six digit code for the user to enter into the app to validate. After entering the valid 2FA, the user will then be redirected to the dashboard to be able to vote. This functionality was accomplished by utilizing NodeJS and node package manager (npm) libraries: qrcode and speakeasy. The QR codes are generated using qrcode and 2FA validation is done through speakeasy. The token that is added by the user is stored in the database to verify that user is indeed is them. This same token is then used to verify that the user has access to the visited page. The token changes every time user logins and application records those changes to keep it up to date for security purposes.

## 4.2 Platform Design

The voting application has been designed with portability and security in mind. The application runs on multiple Docker containers, which makes it deployable from any environment, and would ease the transition of pushing to production, or making the application public. There exist three containers that make up the application:

(1) Reverse Proxy
(2) Application Server
(3) Database

The reverse proxy was built using NGINX, its purpose is to become the interface between a user (outside world) and the actual application. It is able to load balance, and route appropriate requests to the correct application server. This is outside the scope of the project, but it has been added to also assist in using https, or SSL encryption. The webserver can be authorized by a CA, but does require a domain name (which is outside the scope of the project).

The application server was built using the Go language, its purpose is to authenticate, authorize, deliver HTML, route pages, and interact with the database. A user interacts with the application server by making a new user, signing in, changing their information, and casting a vote. A candidate interacts with the application in a similar way. It's the application's job to forward the correct pages to the user, so an example of this is the user being routed to the dashboard page after logging in (on the login page). Read and writes to the database need to come from an authorized party who has been authenticated, which is done by the server. All users create start with no privileges, and are escalated to regular user privileges, such as being able to modify their own information and cast a vote.

The application also handles two-factor authentication, which is required by all users. This means that the user must use Google's authenticator application in conjunction with the voting application to be properly authenticated prior to starting a session, whether to update personal information or to vote.

The database was built using MySQL, its purpose is to store all user, candidate, party, and permissions data. The user will never interact with this directly, and for good reason. Having the database be three layers down, prevents the user from modifying information, or accessing data that they shouldn't. It is solely through the application server, that a user is given permission to make changes the database. The database also stores the hashed passwords of users, which in the case of a data breach, would not allow an attacker to have full access to a user's credentials for logging in.

All specified services and processes outlined above are (as mentioned previously) deployed using Docker containers, where each service is deployed in its own container. This allows them to be able to be run from separate locations, and can be scaled up as needed.

## 5 ANALYSIS

[To Be Completed] The team will analyze the security of the system, evaluate its weak points and create a risk assessment. This will give guidance in the future to the team when refining the security of the system, and determining the potential cost from having an inaccessible website (DDOS attacks) or having user data stolen (data breach), among other cybersecurity attacks. The final report/analysis of the security of the system will allow the team to understand where the system is lacking in security, and give priorities to implementing/changing the system, based on potential cost due to attacks.

## 6 RELATED WORK

TBD

## 7 CURRENT STATUS & FUTURE WORK

The current application currently only runs in local environments, which makes it difficult to execute using SSL encryption due to the lack of utilizing a *real* Certificate Authenticator (CA). This requires local certificates to be generated for every host computer, a tedious task. For this purpose, local deployment requires the bypass of the reverse proxy, or connecting to the application server directly. This means that no load balancing or SSL encryption is used, so despite being able to connect to the server using the https protocol, a warning indicating an invalid certificate will be shown to the user. This is not an issue for local development, but will not be used in production.

## 8 LESSONS LEARNED

TBD

## 9 LEGAL AND ETHICAL CONSIDERATIONS

Publishing the design of the application (as done by this paper) exposes any potential security flaws. This is done for the sole purpose of garnering criticism, to be able to improve the design, and the choice of technologies and techniques used in its implementation. This may also have negative consequences, if an attacker sees a vulnerability and chooses not to disclose that to the team, and instead decides to exploit it before the vulnerability can be found and patched. For this reason, there will need to be a disclaimer to users, they have the right to know that the although the application will continue to improve, opening the design to the public can put their personal information at risk.

## 10 CONCLUSIONS

U.S. elections must be free and fair, Americans must choose their leaders alone, free from interference. That is the purpose of a secure voting application, a system that allows the people to not be eclipsed by the malicious intent of an adversary. The current design has been made public through this document, open to criticism, flexible to change. As the political landscape continuously changes, so must the implementation and practices employed by a secure system. As new threats become known, new security methods and policies need to be added to secure the future.

## REFERENCES

[1] Hunt Allcott and Matthew Gentzkow. 2017. Social Media and Fake News in the 2016 Election. *Journal of Economic Perspectives* 31, 2 (May 2017), 211–36. https://doi.org/10.1257/jep.31.2.211

[2] Max Boot and Max Bergmann. [n.d.]. Defending America From Foreign Election Interference. https://www.cfr.org/report/defending-america-foreign-election-interference

[3] calyptix. 2016. DNC Hacks: How Spear Phishing Emails Were Used. https://www.calyptix.com/top-threats/dnc-hacks-how-spear-phishing-emails-were-used

[4] Philip Ewing. 2019. Facebook, Google And More Meet With Feds To Confer About 2020 Election Security. https://www.npr.org/2019/09/05/757885278/facebook-and-big-tech-meet-with-feds-to-confer-about-2020-election-security

[5] Eric Geller. 2020. Some states have embraced online voting. It's a huge risk. https://www.politico.com/news/2020/06/08/online-voting-304013

[6] Philip N. Howard, Samuel Woolley, and Ryan Calo. 2018. Algorithms, bots, and political communication in the US 2016 election: The challenge of automated political communication for election law and administration. *Journal of Information Technology & Politics* 15, 2 (2018), 81–93. https://doi.org/10.1080/19331681.2018.1448735 arXiv:https://doi.org/10.1080/19331681.2018.1448735

[7] Luke Irwin. 2020. The 5 most common types of phishing attack. https://www.itgovernance.eu/blog/en/the-5-most-common-types-of-phishing-attack

[8] Alice Wenner. 2019. *Securing American Election Report.* Stanford University.

[9] H. Wilcox and M. Bhattacharya. 2016. A framework to mitigate social engineering through social media within the enterprise. In *2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA)*. 1039–1044.