**An Investigation of the Cyber-Attack Susceptibility of State and Local Government Websites**

Thesis

by

Patrick Hill

Submitted to the Graduate School of

Charleston Southern University in fulfillment

of the requirements for the degree of

MASTER OF SCIENCE

April 2021

Advisor:

Yu-Ju Lin, Ph.D.

Table of Contents

[ABSTRACT 3](#_Toc66973194)

[1.0 INTRODUCTION 3](#_Toc66973195)

[2.0 BACKGROUND 3](#_Toc66973196)

[3.0 FINDINGS AND PAST RESEARCH 4](#_Toc66973197)

[3.1 HTTP AND HTTPS ADOPTION 4](#_Toc66973198)

[3.2 TOP LEVEL DOMAIN REGISTRATION 5](#_Toc66973199)

[3.3 VULNERABILITY ANALYSIS 6](#_Toc66973200)

[3.4 TLS USAGE 8](#_Toc66973201)

[4.0 METHODOLOGY 12](#_Toc66973202)

[5.0 CONCLUSION AND FURTHER RESEARCH 13](#_Toc66973203)

[6.0 REFERENCES 15](#_Toc66973204)

# **ABSTRACT**

From an overall national viewpoint, it is important that systems that provide critical data and information to citizens that are own and managed by their local governments are protected from cyber threats. While much research exists that examines internet-facing systems as a whole, the purpose of this study is to provide an analysis on local government systems only in an effort to gauge their overall susceptibility to common threats. By surveying local government-owned, citizen-centric systems, we can better understand the overall rigidity of our nation's cyber framework and its ability to be resilient to cyber-attacks.

# **1.0 INTRODUCTION**

Critical infrastructure is those assets, systems, and networks that underpin American society. State, Local, Tribal, and Territorial (SLTT) websites are a significant part of critical infrastructure. In times of emergency, such as with Covid-19, citizens look to their local government websites for information, guidance, and advisement. This research, "An Investigation of the Cyber-Attack Susceptibility of State and Local Government Websites," will examine more than 2000 public-facing, SLTT government websites to determine cyber-threat susceptibility. Research shows that many SLTT websites, a crucial component of the nation's critical infrastructure, are not secure. While federal guidance, mandates, and regulatory audits have focused upon securing the SLTT infrastructure with scanning, intrusion protection\detection, malware methods, little attention is being placed on the SLTT homepages.

Risk increases when sharing is inconsistent, fragmented, or non-existent. In the spirit of sharing, this study expects to raise awareness of the state, local, tribal, and territorial government website's security status. This research intends to be used as a tool to demonstrate the need for additional federal mandates and to inform internal government site owners and partners of the potential risks discovered in citizen-facing SLTT websites. Managing the looming threats to our nation's cyber-critical systems requires understanding, detection, identification, and collaboration.

In the next section, past research will discuss the background leading to this study on SLTT websites. Next, the paper will provide a literature review with findings that will outline the past research, its importance, and how it applies to SLTT websites. In addition, the findings will present the results of this study, and finally, this research will conclude with a summary of the research and its importance to local governments.

# **2.0 BACKGROUND**

In this section, the needs for this research will be discussed. It is important to understand that a secured and modern government SLTT website uses TLS 1.3, a properly configured E.V. Certificate, patched to mitigate known vulnerabilities, uses HSTS, uses a C.A. in the U.S., and has a .gov domain registration. In addition, it includes disabled and unused ports, such as port 80, when HTTPS is used. Also, the website should support HTTP/2. This research is vital because misinformation and deception can instigate civil disorder. A primary source of citizen information is their local government website. If malicious actors can alter or manipulate our local government websites, then they can manipulate the population.

In 2010 the U.S. government surveyed the cyber landscape and began to take significant steps to improve our resiliency to cyber-attacks. In February 2013, the President issued the Presidential Policy Directive 21 (PPD-21), Critical Infrastructure Security and Resilience, which explicitly calls for an update to the Nation's Infrastructure Plan. In addition, Executive Order 13636: Improving Critical Infrastructure Cybersecurity supplemented PPD-21 by developing, promoting, and incentivizing the adoption of cybersecurity practices and using existing regulation to promote cybersecurity.

As SLTT websites are a crucial tool for citizens, many hacker groups, such as the Russian Internet Alliance or the Iranian Enemies of the People (EOTP), have sought to disrupt citizens' trust and confidence in their governments by targeting SLTT webpages. Malicious actors will try to gain access to an SLTT webpage by exploiting unpatched vulnerabilities, misconfigured ports, or weak encryption and use the compromised web server as a launchpad to a more aggressive cyber campaign. Most cyber-terrorist campaigns intend to disrupt citizen trust and confidence in their governments.

Recently, cybercrime susceptibility has gained increased attention by the U.S. Government due to the impact of state-sponsored hacktivism, hardware\software shipped with spyware, compelled certificates issued for Internet interception, and exploited zero-day exploits.

Since public-facing SLTT government websites are the front-line for delivering official information and accessing local government services, attacks targeting these websites can cause widespread panic and citizen instability. In addition, malicious actors can use compromises to steal citizen credentials and sensitive citizen data. Public-facing website defacements typically involve a cyber threat actor compromising the website or its associated content management system, allowing the actor to upload images to the site's landing page. In situations where such public-facing websites relate to the SLTT government (e.g., the website of a county board of elections), defacements could cast doubt on the websites' information security and legitimacy.

Consider this, if cyber actors could successfully change a state or local government agency website, the underlying data and internal systems would remain uncompromised, yet the website could easily convey false information or show disturbing images. For example, the Town of Hilton Head experienced a breach of their webserver on 10/4/2020, resulting in the below defacement shared with me by the State Law Enforcement Division (SLED). Attacks such as these are not always made public.





Because SLTT websites are considered critical information infrastructures by homeland security, this research will investigate their susceptibility to common cyber-attacks, adopting web security policies such as HTTP Strict Transport Security (HSTS), certificate type, and TLS version supported. In addition, this research will also examine state statutes on data security due to the important role Government policies play in cybersecurity. This research will examine the public external-facing local government agency websites across the U.S. to probe into the concept of our nation's overall cyber threat susceptibility and security. While the literature has been published demonstrating the cause and effect of various exploits, little has been written to show how susceptible our nation's citizen-facing cyberinfrastructure is to these exploits. The Literature review section shows that the current literature on cyber susceptibility focuses on the internet as a whole but not specifically the SLTT government website. This study will apply current research literature on vulnerabilities and test our nation's SLTT web infrastructure.

# **3.0 FINDINGS AND PAST RESEARCH**

This section will provide the findings of this research as well as examples of the past research performed. The past research discussed will examine many insecurities in websites and web communications and how they can be mitigated. How the previous research is essential and how it impacts this study will also be examined.

In addition, this section will provide the findings for HTTP usage, HSTS, HTTP/2, and HTTPS adoption, domain registration, TLS usage, and vulnerability analysis of SLTT websites.

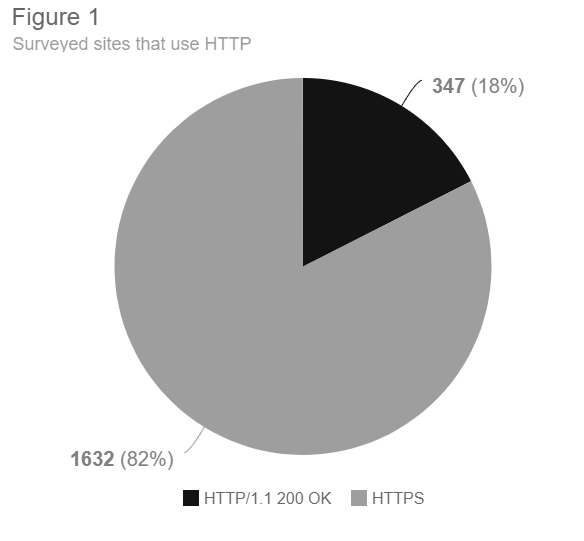
Much research has been done documenting the vulnerabilities and exploits in web systems, yet no recent study has been discovered that has analyzed the U.S. SLTT web domains' susceptibility to cyber threats.

# **3.1 HTTP AND HTTPS ADOPTION**

Many SLTT government websites are not associated with a valid .gov domain and therefore have not gone through the vetting process associated with obtaining a .gov domain. Such few County's use a .gov domain for their website that an attacker can easily spoof a local government site and convey false information to citizens. Minnesota and Texas have the most significant number of county sites that do not use the .gov domain [1]. Poorly secured County websites with a low level of autonomy give attackers a much more realistic opportunity to influence and disrupt citizen activities.

The article "County Election Websites Can Be Easily Spoofed to Spread Misinformation" brings attention to how county government websites in 20 key swing states do not use a .gov domain nor enforce the use of SSL. At the time of the article, Minnesota and Texas have the most significant percentage of non-.gov county government sites, with 95% of their county sites using HTTP [1]. West Virginia, Texas, and Montana have the most considerable number of county governments not using SSL, which would allow attackers to redirect website visitors to alternate, malicious sites. The article states that lack of consistency in website naming and improper use of SSL certificates pose a much more realistic threat to the election process's integrity than a physical attack on voting machines. Often, County election sites are the first-place voters go-to for eligibility requirements, voting locations, registration deadlines, and hours. It is feared that simple misinformation campaigns focused on vulnerable gaps at the local level could negatively impact voting results. Poorly secured county websites give attackers a much more realistic opportunity to influence the outcome of elections. Since not all counties use a .gov domain, voters would have difficulty identifying spoofed sites from real ones [1].

Figure 1 of this research has shown that of the SLTT studied, 18% (347) still accepted HTTP connections. In addition, it is important to note that 95% of SLTT sites that did not support HTTP still had port 80 open.



# **3.2 TOP LEVEL DOMAIN REGISTRATION**

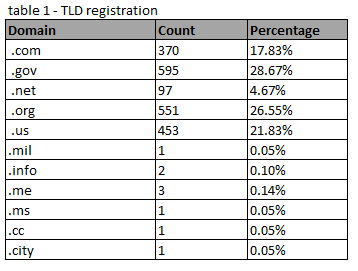
Inconsistency in website naming and the lack of SSL certificate use make county websites a high-profile target for malicious actors. The .gov top-level domain (TLD) facilitates collaboration among government-to-government, government-to-business, and government-to-citizen entities. The TLD authorizes domain names for bona fide US-based government organizations at the federal, state, and local levels, including federally recognized Indian tribes and Alaskan Native groups, known as native sovereign nations (NSNs). The .gov domain designation makes government services easy to identify on the internet.

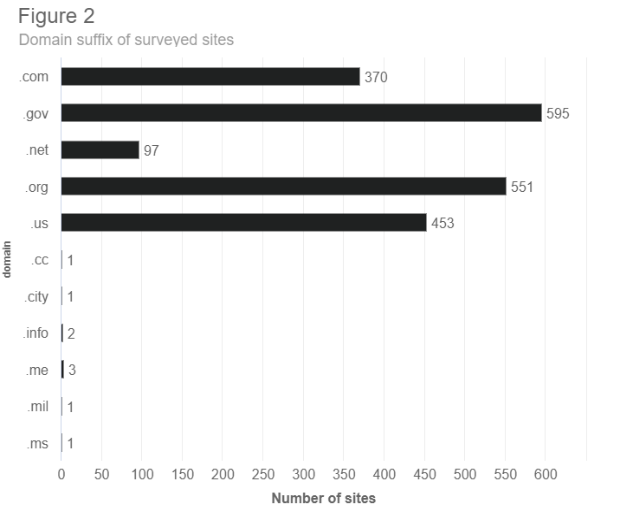
The hacker group APT28 actively interfered with the 2016 presidential election, with many sites being created to spoof local government sites. The article "Microsoft-says-it-has-found-a-Russian-operation-targeting-us-political-institutions" states that during the 2016 election, a group affiliated with the Russian government created fake versions of six websites with the goal of hacking people that visited these fake websites; some of which were related to public policy and the U.S. Senate [3]. U.S. officials repeatedly warned that the November elections are a major focus of malicious interference efforts. APT28, which is sometimes called Strontium or Fancy Bear us a unit under the Russian Military intelligence agency GRU, which specializes in misinformation. APT refers to an advanced persistent threat. Hackers will often send out fake e-mails, directing people to visit sites that appear to be legitimate [3].

Every .gov domain name application is carefully examined to ensure domain names requested will not create misunderstandings about the purpose of domains and their content. The .gov vetting process's overall goal is tomaintain domain name integrity, eligibility is limited to qualified government organizations, and programs for having a managed domain name such as .gov assures citizens that they are accessing an official U.S. government site. General Services Administration (GSA) arbitrates domain name issues and reserves the right to deny domain name requests that do not adequately meet requirements. Title 41 Public Contracts and Property Management in the [Code of Federal Regulations Chapter 102, sub-chapter 173](https://ecfr.federalregister.gov/current/title-41/subtitle-C/chapter-102) outlines Government requirements for a .gov domain. Domain names must be authorized by the Chief Information Officer (CIO) of the requesting or sponsoring governmental organization. For Federal departments and agencies, the General Services Administration (GSA) will accept authorization from the department or agency's CIO. For independent Federal government agencies, boards, and commissions, GSA will accept authorization from the highest-ranking Information Technology Official. For State and local governments, GSA will accept authorization from appropriate State or local officials [11]. On March 10th, 2020 the U.S. began taking notarized signatures for the .gov domain vetting process. Because the U.S. does not require its local governments to have .gov domain registration, official SLTT websites can be easily impersonated. In addition, the financial requirements to obtain a .gov domain are significantly higher than that of the other domain registrations.

Determining the TLD domain registration is an easy process. While our custom CPP program was loading each SLTT domain for analysis, it also collected the domain suffix (.org, .gov, etc.) into table 1.

Based on the study of the 2075 SLTT websites examined, 72% (1480) websites did not use a .gov domain. Table 1 & Figure 2 represent the research breakdown of SLTT domains used.





# **3.3 VULNERABILITY ANALYSIS**

Not only are main local governments using non .gov domains, but they are also slow to adopt HTTPS. HTTPS [3] runs HTTP over Transport Layer Security (TLS), a fundamental security protocol that enables end-to-end encryption and authentication for HTTP connections. Research into a novel attack on TLS titled "A cross-protocol attack on the TLS protocol," authors Nikos Mavrogiannopoulos, Frederik Vercauteren, Vesselin Velichkov, and Bart Preneel show that TLS alone is fallible [4]. The authors present a cross-platform exploit that shows an attacker can interpret signed explicit elliptic curve Diffie-Hellman (D.H.) key exchange parameters as valid plain parameters that enable the impersonation of a trusted server. The server must support the elliptic curve options for the attack to be successful. Proper configuration of HTTPS sites is essential to improving their cyber-security as HTTPS adoption improves. The paper describes the TLS protocol as an agile protocol that allows peers to negotiate their highest supported protocol version and use a combination of ciphers during each session [4]. The TLS cipher suite determines the symmetric encryption cipher with its operational mode, the key exchange method, and the message authentication algorithm. The TLS downgrade dance is for backward compatibility with legacy servers. During handshake negotiation, a server will attempt to use the highest level of TLS that the client supports. If the handshake fails, a retry will be initiated using the next lowest TLS version. POODLE, Drown, and Freak attacks are all examples of attacks against the TLS stack.

**The POODLE attack** (Padding Oracle on Downgraded Legacy Encryption) attack will attempt to intercept the TLS (Transport Layer Security) handshake negotiation to force a weak SSL 3.0 connection. In the paper, "POODLE Bites: Exploiting the SSL 3.0 Fallback," the authors stress the need for proper TLS and webserver security configuration and the disabling of older legacy support protocols such as SSL 3.0 [9]. In the work "SSL and HTTPS: Revisiting Past Challenges and Evaluating Certificate Trust Model Enhancement," authors Clark and Oorschot show many of the on-going security flaws with HTTPS as attacks involving fraudulent certificates, SSL stripping attacks, and the lack of HTTPS support [5]. Not only do attacks against the TLS stack exist but also attacks against the TCP headers. Figure 3 shows the number of sites susceptible to a POODLE attack. Of the SLTT sites examined that accepted TLS connections, 53 sites (3% of the study sample) were susceptible to a POODLE attack.

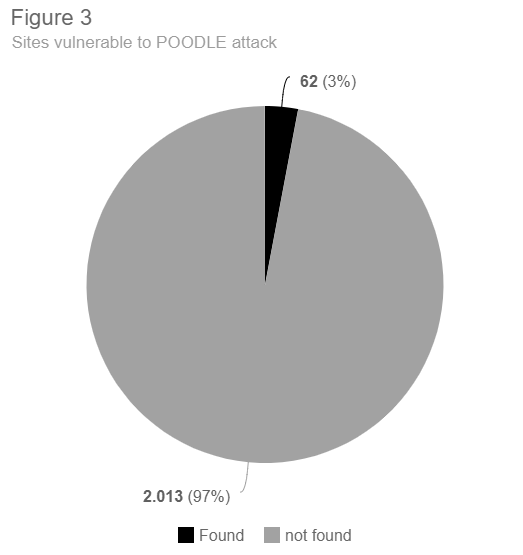
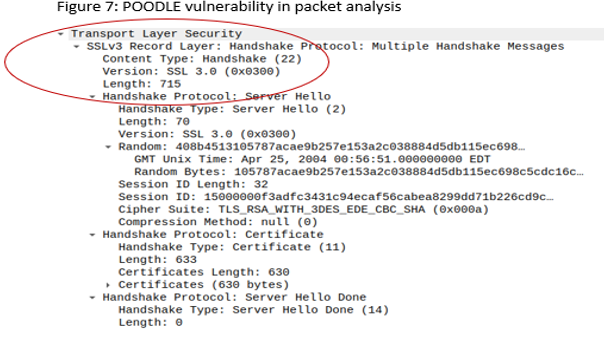
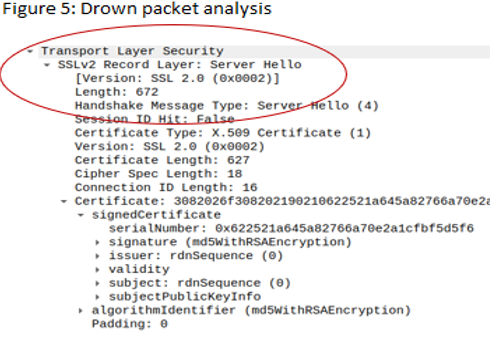


Figure 7 shows an analysis of a packet using a packet analyzer. As we can see in Figure 7, a successful response for the SLTT webserver is accepted utilizing SSL 3.0, making it susceptible to a POODLE attack.

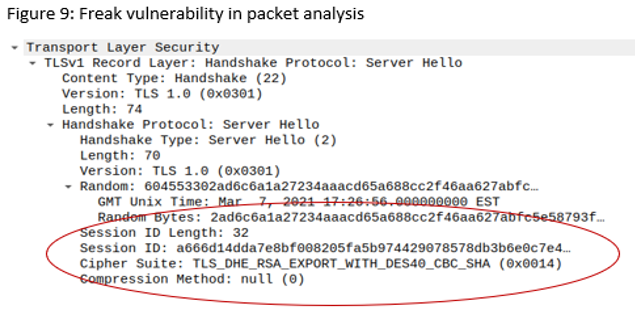


**The Drown attack** (Decrypting RSA with Obsolete and Weakened eNcryption) is like a POODLE attack in that it pushes the server to an obsolete encryption algorithm. DROWN is an attack on an even older protocol version, SSL 2.0. Proper server administration is the key to mitigating this attack; simply disabling support for SSL 2.0 will prevent this vulnerability. Conversely, patching SSL to a later version will also mitigate this attack [18].

Figure 5 shows an analysis of a packet using a packet analyzer. An examination of the ServerHello message reveals a successful connection with SSL 2.0. As indicated by authors Aviram and Schinzel et al., in the research “DROWN: breaking TLS using SSLv2”, any server that accepts a connection on SSL 2.0 is susceptible to the DROWN attack. In addition, the authors point out that any additional several that utilizes the same certificate as a server that uses SSL 2.0, regardless of the TLS version, can be compromised [18].

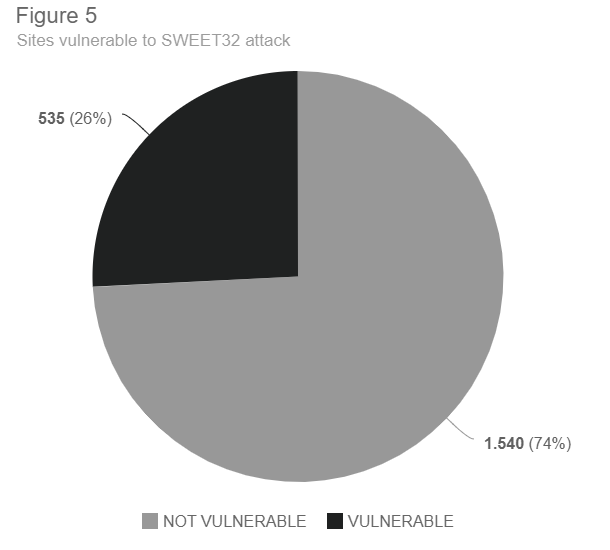
Of the 2075 SLTT websites examined by this research, only 2 (.096%) were susceptible to a DROWN attack.

**The FREAK attack** (Factoring RSA Export Keys) occurs when a man-in-the-middle attacker forces a client to use older and weaker encryption. The attacker will be able to break the encryption and steal sensitive information, including citizen data, or launch an attack by injecting malicious code in the encrypted stream of data. FREAK attacks can be mitigated by patching OpenSSL to the latest version. In the research “A Messy State of the Union: Taming the Composite State Machines of TLS,” the authors used internet-wide scans to estimate that more than alarmingly 25% of HTTPS servers still supported RSA\_EXPORT (FREAK attack) [17]. In Figure 9, the packet analysis reveals a server connection acceptance using an exportable cipher. Of the SLTT websites examined, only four were susceptible to a FREAK attack.

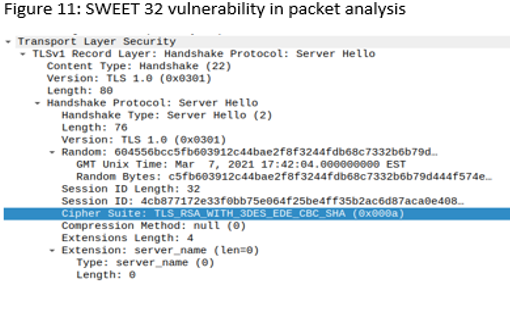


Of the 2075 SLTT websites examined, only 4 (.19%) were susceptible to a FREAK attack.

**The SWEET32 attack** is an attack on block ciphers that have a block size of 64 bits. These ciphers are vulnerable to a practical collision attack when used in CBC mode. This attack is easily mitigated by deprecating all versions of SSL/TLS protocols that support cipher suites that use 3DES as the symmetric encryption. The DES and Triple DES ciphers, as used in the TLS, SSH, and IPSec protocols and other protocols and products, have a birthday bound of approximately four billion blocks, which makes it easier for remote attackers to obtain cleartext data via a birthday attack against a long-duration encrypted session, as demonstrated by an HTTPS session using Triple-DES in CBC mode, aka a “Sweet32” attack. Figure 5 shows that of the SLTT websites examined on 535 (26%) were using legacy block ciphers and there susceptible to the Sweet32 attack.



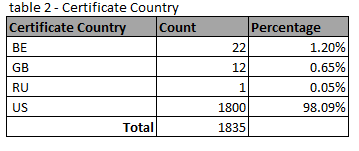
Older ciphers, such as TLS\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA, lack of [forward](https://en.wikipedia.org/wiki/Forward_secrecy) secrecy, meaning there is no encrypted key exchange. Therefore, if the private RSA key is leaked, all passively captured past data exchanges can be decrypted. In addition, the CBC padding issues are available to attackers because AES-CBC is used in a mac-then-encrypt situation where the padding is removed before the message is authenticated. Figure 11 shows the packet details of a site susceptible to a SWEET32 attack. Based on this study, it was discovered that no SLTT website was susceptible.

**Heartbleed** is a code flaw in the OpenSSL cryptography library. In the study, “The Matter of Heartbleed” [10], the authors perform a comprehensive, measurement-based analysis on the vulnerability impact, including tracking the population, monitoring the patching behavior over time, and assessing the HTTPS ecosystem's impact,￼￼￼￼￼￼￼ ￼an￼d￼ exposing real attacks. The authors found that 44 of the top 100 Alexa websites remain vulnerable two months after the patch was released. In addition, only 10% of analyzed websites replaced their certificates compared to 73% that patched their ￼site￼￼￼ and 14% of those using the same private key [10]. HTTPS i￼￼￼￼s the secu￼￼￼￼￼￼re￼ variant of the HTTP protocol on which the web is based. HTTPS provides cryptographic security protections by carrying HTTP messages over the TLS protocol instead of directly over TCP (Transmission Control Protocol). HTTPS websites authenticate using digital certificates as part of the TLS handshake. Web users are shown an invalid certificate warning when their browser cannot validate the identity of the websites they are visiting. Based on this study, no SLTT websites examined were susceptible to a Heartbleed attack.

While these warnings often appear in benign situations, they can also signal a man-in-the-middle attack. However, many more frequent users are connecting to a legitimate website with erroneous or self-signed certificates. The research performed by Joshua Sunshine, Serge Egelman, Hazim Almuhimedi, Neha Atri, and Lorrie Faith Cranor in "Crying Wolf: An Empirical Study of SSL Warning Effectiveness" [7] shows that invalid certificate warnings can signal a man-in-the-middle attack or a DNS (Domain Name System) spoofing attack. The authors surveyed 400 Internet users to examine their reactions to understanding website certificate warnings and their effectiveness. Their research showed that the warnings are often by-passed and that preventing users from making connections to unsafe websites is the safer approach [7].

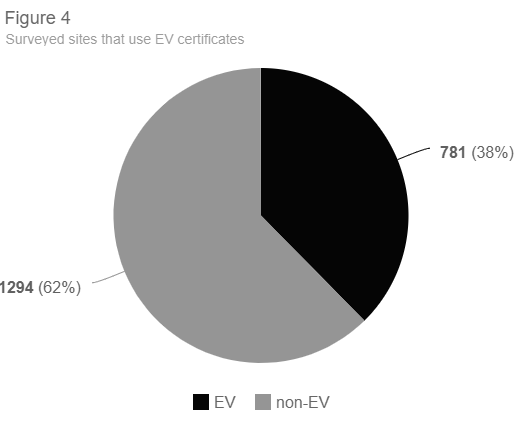
POODLE, FREAK, DROWN, SWEET32, and HEARTBLEED vulnerabilities and exploits demonstrate the importance of thorough website administration as they can easily be mitigated by disabling older, more insecure versions of SSL or by patching servers with the latest code updates.

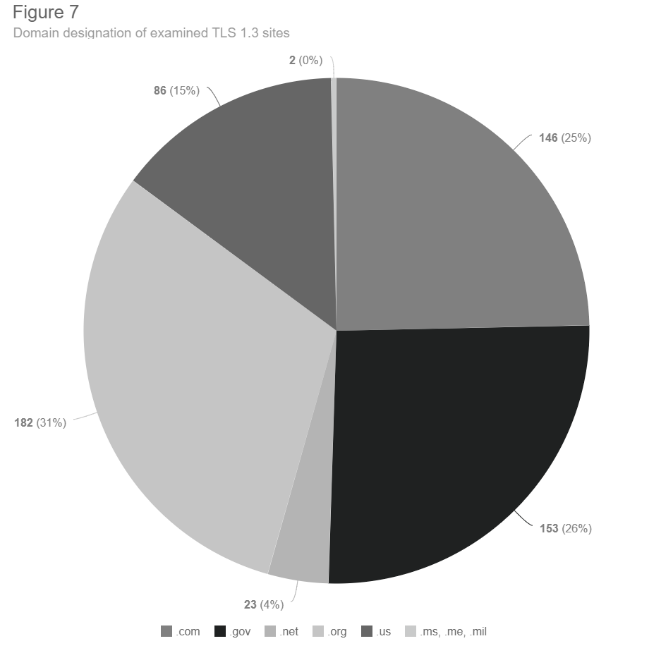
# **3.4 TLS USAGE**

Browsers also enforce additional policies for HTTPS pages, for example, ensuring that HTTPS pages cannot load scripts from non-secure sources. ￼Authors Felt, Barnes, et al. of the paper "Measuring HTTPS Adoption on the Web" attempt to measure the HTTPS adoption rate [12].￼ The authors st￼￼ate the tremendous growth in HTTPS adoption has been positively trending since 2016, with half of the top 100 websites supporting HTTPS. HTTPS provides cryptographic security protections by carrying HTTP messages over TLS instead of directly over TCP. HTTPS works by utilizing public-key cryptography and third-party digital signatures to encrypt and confirm data communications between clients and servers. Communications between the browser and the webserver are not accessible in plaintext to intermediate entities. Intermediate entities cannot make modifications to content sent between the browser and the webserver. The client is assured that the other end of the channel is the one that it intends to communicate with [12]. Projects like the "Let's Encrypt," HTTPS only standard and search ranking changes to promote HTTPS are credited with pushing HTTPS adoption rates. Browsers now require HTTPS to unlock and enable certain features. HTTPS is focused on protection against network attackers but does not protect against other types of attacks. However, without HTTPS, the job of attackers becomes much more accessible. Without HTTPS, traffic can be intercepted by an inline adversary, as well as off-path adversaries who are capable of hijacking routes. This inflight manipulation of web page content is one of the significant reasons to abandon HTTP websites. 

Nearly all secure web communications take place over HTTPS. HTTPS is based on TLS encrypted transport protocol and supporting key infrastructure of thousands of certificate authorities (C.A.'s) – entities trusted by users' browsers to vouch for the identity of a webserver. TLS is one of the major secure communication protocols on the internet. It is an agile protocol that allows peers to negotiate their highest supported protocol version and the combination of ciphers used in a session. Not all cipher suites within TLS are strong. The paper "Analysis of the HTTPS Certificate Ecosystem" lists that data is collected by performing 110 Internet-wide scans over 14 months and identify vulnerabilities and user-facing errors that negatively impact the internet ecosystem's overall security [8]. In the paper, the authors investigate the trust relationship between root authorities. The authors analyzed 1832 CA certificates controlled by 683 organizations and found that 80% of the organizations do not have commercial certificate authorities’ certificates. The CA's constraints investigated and found that only 7 C.A. (Certificate Authority) certificates use name contrasts, and more than 40% of the C.A.'s have no length constraint. The authors identify two sets of mis-issued C.A. certificates. The authors found many problematic security issues within their study, such as a public key compromise that would require 26% of the HTTPS websites to obtain new certificates. They found that half of the trusted leaf certificates contain an inadequately secure 1024 RSA key in their trust chains. Based on this study, 29% of all certificates were issued by ‘Let’s Encrypt’ with Sectigo occupying second place with 18%. Only 1% of certificates were self-signed. Table 2 shows the distribution of Country codes between 1835 SLTT websites examined, with 98% of sites having C.A.’s that originated from the U.S.

However, malicious actors can undermine the trustworthiness of the TLS certificates framework. While expired and self-signed certificates can encrypt data, the security alerts displayed will cause citizens to believe the data is untrustworthy. A standard, domain-based validation provides a low-security level, as a man-in-the-middle adversary can impersonate a domain can obtain a trusted domain validated (DV) certificate for this domain. C.A.s (Certificate Authorities) offer distinct types of certificates to remedy such problems and meet stronger protection demands. The types differ in the way a C.A. conducts the identity validation. Extended Validation (E.V.) certificates are believed to be more secure since a C.A. completing the public key validation is obligated to perform a more detailed validation, sometimes including a face-to-face verification. The extended validation (E.V.) certificates are believed to be the most secure of the certificate offerings because the C.A. must conduct a rigorous identity verification procedure [15]. Figure 4 shows the number of E.V. certificates in use at the time of the survey. Of the 2075 sites surveyed, 38% (781) use E.V. certificates.



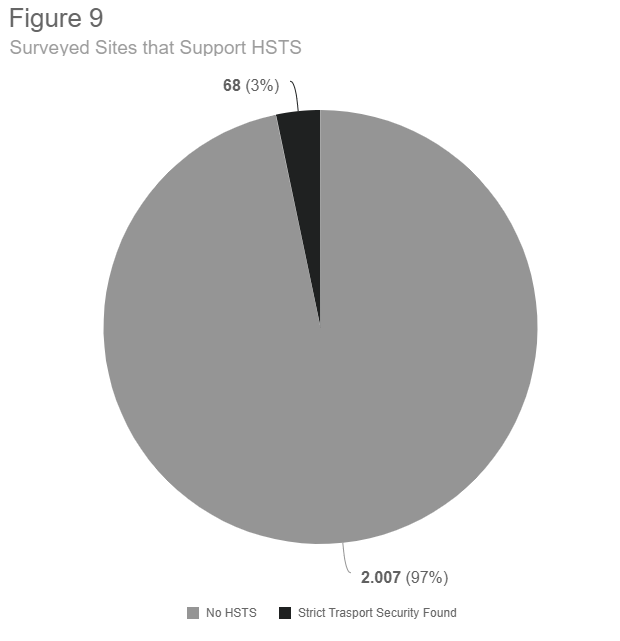
Content Providers (CP) are servers that host web resources for websites. When a website loads resources from untrusted servers, various undesirable consequences can occur, such as the execution of malicious scripts, malicious content, mining cryptocurrencies, or sending bot-net attacks [8].  In the research work “Re-Architecting the Internet” [6], the authors state that security in the WWW architecture is based on authenticating the source server and securing the data during transport without considering the content itself. The traditional assumption is that a page is as secure as the server hosting it. However, modern websites often have a composite structure where different actors author components of the web page, and one logical page contains components collected from disparate servers. Applying a single security policy to a whole page is inadequate. They introduce a novel way of protecting users from web-based malware, which is a new model that uses opportunistic personas to better secure web content by adding integrity and accountability to individual elements. In this paper, the authors present the overall design of the mechanism and details derived from a prototype of the system [6].

**TLS 1.3** is an important enhancement because it eliminates several vulnerabilities. The Transport Layer Security (TLS) Protocol Version 1.3 as of this writing, TLS 1.3 is the latest version of TLS. This new protocol version is almost a complete redesign, with striking differences to previous versions in the protocol and its use of cryptography. In the research “Tracking the deployment of TLS 1.3 on the Web: A story of experimentation and centralization” performed by Holz and Hiller et al., the authors note that just 15 months after standardization, it is used in about 20% of connections they observed. Deployment on popular domains is at 30% and at about 10% across the com/net/org top-level domains (TLDs) [19]. This research has revealed that 29% (592) of examined SLTT websites supported TLS 1.3. In addition, Figure 7 shows that 26% of the TLS 1.3 adopted SLTT sites were .gov domain name registrations.

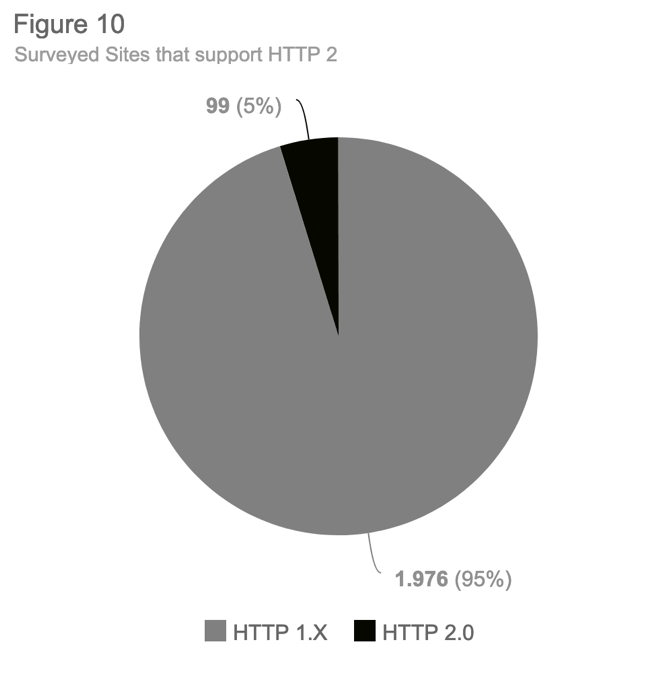
TLS 1.3 eliminates the padding in block ciphers. Block ciphers are still accepted, but they must be run in stream mode, which creates streams of pseudo-random data of arbitrary length. In addition, encryption and authentication have been combined into a single element. This new type of variant of Authenticated Encryption construction is called Authenticated Encryption with Additional Data (AEAD). Finally, TLS 1.3 only includes the bulk cipher and the hashing algorithm TLS\_AES\_256\_GCM\_SHA384. Separate key exchange and signature algorithms are no longer needed. Also, TLS 1.3 includes just five recommended cipher suites:

* TLS\_AES\_256\_GCM\_SHA384
* TLS\_CHACHA20\_POLY1305\_SHA256
* TLS\_AES\_128\_GCM\_SHA256
* TLS\_AES\_128\_CCM\_8\_SHA256
* TLS\_AES\_128\_CCM\_SHA256

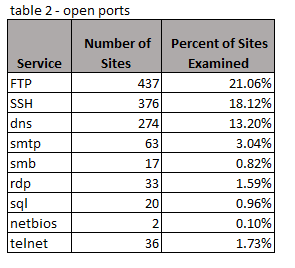
The operational benefits of being able to control both endpoints of a connection are undeniable. My research observed that 29% of all SLTT examined accepted TLS 1.3 connections. While these sites did not accept TLS 1.3 only, the adoption of TLS 1.3 is an exceptional start.

HTTP Strict Transport Security (HSTS) is a policy for web security in which a web server￼￼ can restrict communications only to HTTPS connections from browsers, denying HTTP connections. HSTS helps mitigate man-in-the-middle attacks by pr￼￼eventing clients from downg￼￼rading from HTTPS to HTTP. An HTTP request to a website enabled with HSTS will automatically be upgraded to HTTPS. Figure 9 shows that based on the research performed against 2075 SLTT websites, only 3% (68) supported HSTS.

**HTTP/2** is the latest iteration of the HTTP protocol. In the research "Exploring HTTP Header Manipulation In-the-Wild" [16], the authors investigated web page headers' exploitation. Headers are attribute-value pairs that are emended within all HTTP messages. Headers are a critical complement to HTTP; their investigation reveals that 25% of their measured autonomous systems (AS) modify the HTTP headers. Once headers are transmitted across a network, they become vulnerable to manipulation. Therefore, a secure site should use the most current iteration of HTTP headers, HTTP/2.0, which is currently supported by all major browsers. HTTP2 has many performance enhancements over the HTTP1 standard, but to utilize these improvements, TLS is required. This means that TLS encryption is mandatory for a website to take advantage of HTTP/2's performance advantages [16].

Outside of the performance benefits of the protocol, HTTP/2 improves security by requiring TLS [20]. The research conducted shows that out of 2075 sites examined, only 5% (99) accepted connection from HTTP/2 (Figure 10). 

**Open ports** on a web server that is internet facing pose a serious risk to the web server itself. Open ports are not dangerous when the service listening on the port is misconfigured, unpatched, or vulnerable to exploitation. In addition, malicious services can use open ports to communicate and exfiltrate data without using traditional ports and can add another layer of detection avoidance. Based on the sites researched, 40% (831) web servers had ports open other than 80 and 443, the traditional web traffic ports (Figure 11). Less open ports will reduce the attack surface of the webserver. Some of the riskiest ports to leave open are FTP, SSH, Telnet, SMTP, DNS, SQL, SMB, NetBIOS, and RDP. Table 2 shows the breakdown of services open on 2075 examined SLTT websites.

File Transport Protocol (FTP) is an outdated and insecure protocol and uses port 21. FTP does not utilize any encryption for both data transfer and authentication. Therefore, it is not recommended that the service is used, and the port should be disabled. Based on the research, over 21.06% (437) of the SLTT sites examined had port 21 enabled.

Secure Shell (SSH) is used for remote management and utilizes port 22. While it is generally considered secure, it requires proper key management and configuration. It is not recommended the port 22 is accessible via the internet. Based on the research, over 18.12% (376) of the SLTT sites examined had port 22 enabled.

Telnet, the predecessor to SSH, utilizes port 23. While it is no longer considered secure and is frequently abused by malware. Based on the research, over 1.73% (36) of the SLTT sites examined had port 23 enabled.

Simple Mail Transport Protocol (SMTP) utilizes port 25 and is used for e-mail. If not properly secured, it can be abused for spam e-mail distribution. It is not recommended that an SLTT web server should also be used as an e-mail distribution system. Therefore, the port should be disabled. Based on the research, over 3.04% (63) of the SLTT sites examined had port 25 enabled.

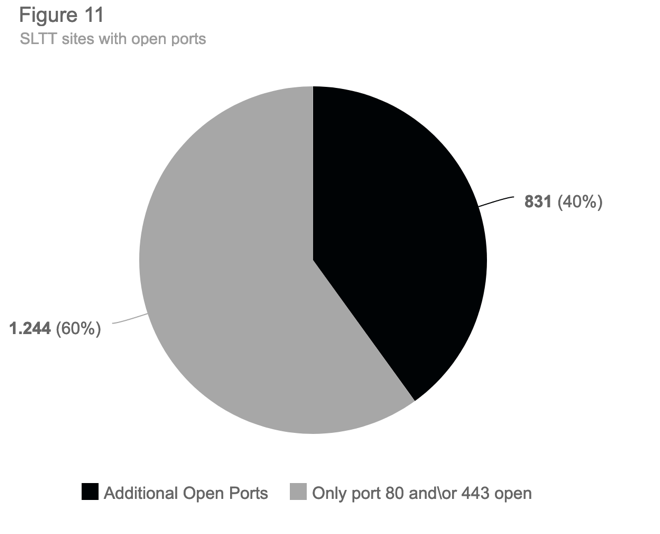
Domain Name Service (DNS) is often used for DDoS attacks and utilizes port 53. It is recommended that an SLTT webserver should not perform double-duty as a DNS server as well. Therefore, port 53 should be disabled. Based on the research, over 13.20% (274) of the SLTT sites examined had port 53 enabled.

NetBIOS, a legacy protocol that uses port 13,9 is primarily used for file and printer sharing. An SLTT website should not be using NetBIOS for sharing service; therefore, the port should be disabled. Based on the research, over .10% (2) of the SLTT sites examined had port 139 enabled.

Server Message Blocks (SMB) uses port 445 for its sharing capabilities of files and printers. SMB was exploited in 2017 by the famous WannaCry attack. Therefore, port 445 should be disabled and an SLTT internet-facing webserver should not be used for file and print services. Based on the research, over .82% (17) of the SLTT sites examined had port 445 enabled.

The default ports for Structured Query Language (SQL) are ports 1433,1434, and 3306. These ports are easily able to be used for malware distribution. Based on the research, over .96% (20) of the SLTT sites examined had the default SQL enabled.

Remote Desktop (RDP) uses port 3389 and is arguably one of the riskiest ports to have open on an internet-facing server. It is utilized to exploit various vulnerabilities in remote desktop protocols, as well as weak user authentication. Remote desktop vulnerabilities are commonly used in real-world attacks, with the last example being the BlueKeep vulnerability. Based on the research, over 1.59% (33) of the SLTT sites examined had port 3389 enabled.



In summary of this section, informed decision-making requires the ability to discover, retrieve and use accurate and actionable information to drive results. Past research has identified various weak points and threats facing the underlying operations of our web structures. Research has shown that HTTPS is important because malicious 3rd parties can intercept network traffic without it. Research has shown that E.V. certificates are designed to prevent phishing attacks and are more secure due to the rigorous C.A. checks that are required to obtain them. In addition, self-signed certificates are problematic due to the errors on the client-side then generate which leads to citizen distrust. Certificate county of origin is important as well. Relying on certificate authorities that are based outside the U.S. can lead to exploitation of the TLS communication channel by foreign powers. Emerging technology, such as HSTS, has demonstrated their importance because it allows a server to specify that it will only accept HTTPS connections from browsers, preventing man-in-the-middle attacks. User confidence can be increased by applying and implementing a .gov domain to the SLTT site. Additionally, attacks against the TLS stack are easily mitigated with proper site maintenance and patching. My research shows that many SLTT websites have not corrected these published vulnerabilities, nor have they adopted best practices or implemented emerging security features.

# **4.0 METHODOLOGY**

In this section, I will provide the methodology I used to conduct my research. Threats and vulnerabilities should be identified and managed in a coordinated and comprehensive way. There is no sole source that maintains the list of local government websites; therefore, I will manually create a dataset of more than 2075 SLTT web domains.

The research performed examines and logs 2075 U.S. state and local government websites for HTTPS adoption, HSTS adoption, valid certificates, originating certificate county, and TLD domain designation, highest TLS version and vulnerability status.

I used the below combination of custom code and standard analysis tools to gather the data from the SLTT websites.

For this research I used the OpenSSL command below to gather Certificate Country Code, TLS version, Certificate Issuer, Certificate type and Certificate signed\expiration date.

**echo "Q" | openssl s\_client -connect <SLTT website> 443.**

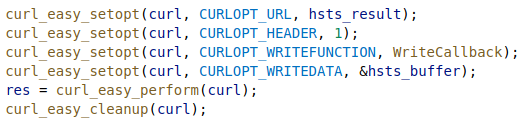
For gathering vulnerability and exploit analysis, open ports, TLS version, and certificate issuer this research uses the below NMAP command.

**nmap -p 80,443 --script=ssl-enum-ciphers, ssl-cert, ssl-poodle, ssl-heartbleed, sslv2-drown**

**<SLTT website>**

This research uses the below CURL command for determining the HTTP, HTTPS, and HSTS adoption.

Curl:



The GitHub link below will lead to the complete code used for this research.

[**https://github.com/patrickjhill01/thesis**](https://github.com/patrickjhill01/thesis)**.**

Additionally, I have contacted local government agencies of sites that are the most susceptible to attack to determine if site administration is done 'in-house,' outsourced or if a non-IT on staff individual develops and maintains the website.

My research uses a combination of personnel inquiry and standard I.T. Admin tools such as Nmap, OpenSSL, and curl to catalog the various aspects of susceptibility. Nmap, OpenSSL, and Curl were selected because they are readily available and are a part of the typical tools available within a typical I.T. professionals toolbox. In addition, the selected tools provide little impact to the SLTT webserver’s functionality and do not hinder a sites performance.

# **5.0 CONCLUSION AND FURTHER RESEARCH**

To conclude, the background has shown that SLTT web systems are considered by the United States as critical infrastructure and therefore, are a significant target by malicious threat actors. This study has shown that our nations state, local, tribal, and territorial (SLTT) websites are not as secure and as well maintained as they should be. The methodology used to prove this study is gathered with Nmap, Curl, OpenSSL, and personal query.

New technologies such as TLS 1.3, HSTS, Enhanced Validation Certificates, and HTTP/2 are not widely adopted by SLTT websites. The proper .gov domain registration, one of the critical components of instilling citizen confidence, is only used by 28% of the SLTT sites examined. While 70% of SLTT sites had certificates signed by U.S. organizations, 30% were from outside the U.S. I.T. house-keeping items such as the use of self-signed or expired certificates seemed to have been resolved and afflict only >1% of the sites examined but 60% of SLTT websites had additional risky server ports open and exposed to the internet. In addition, 33 SLTT sites had remote desktop protocol (RDP) port open to the internet.

Legacy attacks such as POODLE, Heartbleed, DROWN, and FREAK have been mitigated mainly with the exception of SWEET32. There are 535 SLTT sites that are still susceptible to the SWEET32 attacks. Additional work is needed by SLTT admins.

The literature review that I have presented has shown how malicious attacks against the TLS\SSL stack can be mitigated with proper patching and site administration, as well as the importance of E.V. certificates, HSTS, TLS 1.3, C.A. (Certificate Authority) country origin, HTTP/2 adoption, and domain registration.

Because public-facing local government websites are the front-line for delivering official information (such as vaccine information) and accessing local government services, attacks targeting these websites can cause widespread panic and citizen instability. In addition, a poorly secured government site can be used to steal citizen credentials, expose sensitive citizen data, and act as a pivot point for malicious threat actors. This research is vital in our understanding our susceptibility of cyber-attacks against our state and local governments. This research shows that SLTT websites are not as secure as they should be and that citizen confidence, data and stability are at risk.

Further research is needed to examine additional SLTT supplementary sites, such as law enforcement, hospital and court sites. Also, more research needs to be performed to track the remediation rates for the SLTT systems over time. This will aid our wholistic understanding of our cyber-attack susceptibility as well as provide evidence for a need for federal mandates for cyber security guidelines aimed at all types of citizen facing government websites.

# **6.0 REFERENCES**

1. Vijayan, Jai (2018, March 23) *County Election Websites Can Be Easily Spoofed to Spread misinformation.* DarkReading.

<https://www.darkreading.com/vulnerabilities-threats/county-election-websites-can-be-easily-spoofed-to-spread-misinformation/d/d-id/1333132>

1. Hsu-Chun Hsiao, Tiffany Hyun-Jin Kim, Yu-Ming Ku, Chun-Ming Chang, Hung-Fang Chen, Yu-Jen Chen, Chun-Wen Wang, and Wei Jeng. 2019. *An Investigation of Cyber Autonomy on Government Websites*. In The World Wide Web Conference (WWW '19). Association for Computing Machinery, New York, NY, USA, 2814–2821. DOI:https://doi.org/10.1145/3308558.3313645
2. Dwoskin, E. & Timberg, C (2018, August) *Microsoft says it has found a Russian operation targeting U.S. political institutions.* Washington Post.

<https://www.washingtonpost.com/business/economy/microsoft-says-it-has-found-a-russian-operation-targeting-us-political-institutions/2018/08/20/52273e14-a4d2-11e8-97ce-cc9042272f07_story.html?noredirect=on&utm_term=.db741731087b>

1. Nikos Mavrogiannopoulos, Frederik Vercauteren, Vesselin Velichkov, and Bart Preneel. 2012. *A cross-protocol attack on the TLS protocol.* In Proceedings of the 2012 ACM (Association for Computing Machinery) conference on Computer and communications security (CCS '12). Association for Computing Machinery, New York, NY, USA, 62–72. DOI:https://doi.org/10.1145/2382196.2382206
2. J. Clark and P. C. van Oorschot, "SoK: SSL and *HTTPS: Revisiting Past Challenges and Evaluating Certificate Trust Model Enhancements,*" 2013 IEEE Symposium on Security and Privacy, Berkeley, CA, 2013, pp. 511-525, doi: 10.1109/SP.2013.41.
3. Anja Feldmann, Luca Cittadini, Wolfgang Mühlbauer, Randy Bush, and Olaf Maennel. 2009. *HAIR: hierarchical architecture for internet routing. In Proceedings of the 2009 workshop on Re-architecting the internet* (ReArch '09). Association for Computing Machinery, New York, NY, USA, 43–48. DOI:https://doi.org/10.1145/1658978.1658990
4. Joshua Sunshine, Serge Egelman, Hazim Almuhimedi, Neha Atri, and Lorrie Faith Cranor. 2009. *Crying wolf: an empirical study of SSL warning effectiveness.* In Proceedings of the 18th conference on USENIX security symposium (SSYM'09). USENIX Association, USA, 399–416.
5. *Z*akir Durumeric, James Kasten, Michael Bailey, and J. Alex Halderman. 2013. *Analysis of the HTTPS certificate ecosystem*. In Proceedings of the 2013 conference on Internet measurement conference (IMC '13). Association for Computing Machinery, New York, NY, USA, 291–304. DOI:https://doi.org/10.1145/2504730.2504755
6. Möller, Bodo; Duong, Thai & Kotowicz, Krzysztof. *This POODLE Bites: Exploiting the SSL 3.0 Fallback*, report, September 2014; (<https://digital.library.unt.edu/ark:/67531/metadc949502/>: accessed November 12, 2020), University of North Texas Libraries, UNT Digital Library, [https://digital.library.unt.edu](https://digital.library.unt.edu/); crediting UNT Libraries Government Documents Department.
7. Zakir Durumeric, Frank Li, James Kasten, Johanna Amann, Jethro Beekman, Mathias Payer, Nicolas Weaver, David Adrian, Vern Paxson, Michael Bailey, and J. Alex Halderman. 2014. *The Matter of Heartbleed.* In Proceedings of the 2014 Conference on Internet Measurement Conference (IMC '14). Association for Computing Machinery, New York, NY, USA, 475–488. DOI:https://doi.org/10.1145/2663716.2663755
8. govinfo. (2021). Code of Federal Regulations Chapter 102, Sub-Chapter 173. https://www.govinfo.gov/app/details/CFR-2010-title41-vol3/CFR-2010-title41-vol3-part102-id2024
9. Adrienne Porter Felt, Richard Barnes, April King, Chris Palmer, Chris Bentzel, and Parisa Tabriz. 2017. *Measuring HTTPS adoption on the web*. In Proceedings of the 26th USENIX Conference on Security Symposium (SEC'17). USENIX Association, USA, 1323–1338.
10. govinfo. (2021). Code of Federal Regulations Chapter 102, Sub-Chapter 173. <https://www.govinfo.gov/app/details/CFR-2010-title41-vol3/CFR-2010-title41-vol3-part102-id2024>
11. Executive order -- improving critical infrastructure cybersecurity. (n.d.). Retrieved February 11, 2021, from <https://obamawhitehouse.archives.gov/the-press-office/2013/02/12/executive-order-improving-critical-infrastructure-cybersecurity>
12. Pawel Szalachowski. 2019. EVLA: Extended-Validation Certificates with Location Assurance. In Proceedings of the 2019 ACM International Symposium on Blockchain and Secure Critical Infrastructure (BSCI '19). Association for Computing Machinery, New York, NY, USA, 73–79. DOI:https://doi-org.mendel.csuniv.edu/10.1145/3327960.3332379
13. Gareth Tyson, Shan Huang, Felix Cuadrado, Ignacio Castro, Vasile C. Perta, Arjuna Sathiaseelan, and Steve Uhlig. 2017. Exploring HTTP Header Manipulation In-The-Wild. In Proceedings of the 26th International Conference on World Wide Web (WWW '17). International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 451–458. DOI:https://doi-org.mendel.csuniv.edu/10.1145/3038912.3052571
14. Benjamin Beurdouche, Karthikeyan Bhargavan, Antoine Delignat-Lavaud, Cédric Fournet, Markulf Kohlweiss, Alfredo Pironti, Pierre-Yves Strub, and Jean Karim Zinzindohoue. 2017. A messy state of the union: taming the composite state machines of TLS. Commun. ACM 60, 2 (February 2017), 99–107. DOI:https://doi-org.mendel.csuniv.edu/10.1145/3023357
15. Nimrod Aviram, Sebastian Schinzel, Juraj Somorovsky, Nadia Heninger, Maik Dankel, Jens Steube, Luke Valenta, David Adrian, J. Alex Halderman, Viktor Dukhovni, Emilia Käsper, Shaanan Cohney, Susanne Engels, Christof Paar, and Yuval Shavitt. 2016. DROWN: breaking TLS using SSLv2. In Proceedings of the 25th USENIX Conference on Security Symposium (SEC'16). USENIX Association, USA, 689–706.
16. Ralph Holz, Jens Hiller, Johanna Amann, Abbas Razaghpanah, Thomas Jost, Narseo Vallina-Rodriguez, and Oliver Hohlfeld. 2020. Tracking the deployment of TLS 1.3 on the web: a story of experimentation and centralization. *SIGCOMM Comput. Commun. Rev.* 50, 3 (July 2020), 3–15. DOI:https://doi-org.mendel.csuniv.edu/10.1145/3411740.3411742
17. Konrad Wolsing, Jan Rüth, Klaus Wehrle, and Oliver Hohlfeld. 2019. A performance perspective on web optimized protocol stacks: TCP+TLS+HTTP/2 vs. QUIC. In *Proceedings of the Applied Networking Research Workshop* (*ANRW '19*). Association for Computing Machinery, New York, NY, USA, 1–7. DOI:https://doi-org.mendel.csuniv.edu/10.1145/3340301.3341123