Investigating Secure Implementation of Government Web based Systems in Tanzania

Aron Kondoro
University of Dar es Salaam
Dar es Salaam
Tanzania

Joel Mtebe
University of Dar es Salaam
Dar es Salaam
Tanzania

ABSTRACT
The government of Tanzania has been adopting various web-based systems to improve public services to its citizens. With these systems being online, security and privacy have started to play a key role. Many systems use HTTP over Transport Layer Security (HTTPS) to secure their web front ends. However, many HTTPS implementations still suffer from several security and privacy problems. This study investigated the security of HTTPS implementations government web-based systems in Tanzania. Using a sample of 74 government web-based systems, an automated tool testssl was used to check for well-known HTTPS/SSL vulnerabilities, configuration mistakes, support for outdated and vulnerable protocols, and adherence to HTTPS best practices. Results show that 43% of web systems have serious HTTPS security issues due to vulnerabilities, and configuration mistakes. These issues can lead to system com- promise, disclosure of sensitive information, and loss of privacy to citizens. The study highlights these security issues that may have been overlooked and offers suggestions that may prevent them in the future

Keywords
Web Security, HTTPS, TLS/SSL, e-Government

1. INTRODUCTION
The government of Tanzania has been increasingly adopting web-based systems to improve the delivery of public services to its citizens. These systems have allowed the government to interact with its citizens via the Internet. This has lowered operation costs as well as widened access to government services. Some examples of web-based systems that have been implemented include the Tanzania Revenue Authority (TRA) system which allows citizens to register for tax payments and submit tax returns, the Tanzanian Police Force System which allows the public to report crimes, and the Higher Education Student's Loan Board (HESLB) which allows students to apply for university loans.

With these systems and many others being online, security and privacy have been one of the main issues that have taken center stage. Users who interact and log into these systems are in a constant risk from various cyber-attacks that exist online. Sensitive information such as usernames and passwords exchanged between these systems and their users can be compromised and used for malicious purposes. Digital identities can also be stolen from users to impersonate them and perform illegal acts on their behalf. Furthermore, online activities of users can potentially be monitored, leading to privacy concerns over personal information. In response to this, the government has started to consider various ways of improving the security and privacy of these systems.

Under the National Information and Communication Technology (ICT) policy of 2016 [20] the Tanzanian government provided directives on the security and privacy of its ICT systems. Government agencies and other public institutions that deploy web-based systems have been tasked to ensure that their online services are provided in a trusted and secure environment. As a result, they have started using the HTTP over Transport Layer Security (HTTPS) protocol to protect the content and communication of their web-based systems.

Despite these efforts, implementing HTTPS in a secure way remains a challenging endeavor. This is due to the constant emergence of new vulnerabilities and attacks. For instance, recently, there have been several high-profile security attacks against HTTPS such as the Padding Oracle on Downgraded Legacy Encryption (POODLE) attack which can allow an attacker to intercept secure communication by taking advantage of SSL 3.0 vulnerabilities [21]. Moreover, there have been other various security attacks such the Browser Exploit Against SSL/TLS (BEAST) [11], Compression Ratio Info-leak Made Easy (CRIME) [27], Browser Reconnaissance and Exfiltration via Adaptive Compression of Hypertext (BREACH) [13], and Heartbleed [5] which all have compromised the security of HTTPS. As a result, the HTTPS protocol has gone through several revisions to fix these problems.

Nevertheless, many government web systems still fail to update their HTTPS implementations. Once HTTPS has been configured in these systems, they tend to be forgotten and not properly maintained. Consequently, a lot of HTTPS vulnerabilities that can be easily exploited continue to exist. In addition, personnel who maintain these systems frequently make implementation mistakes or omit important HTTPS parameters during configurations that leave security holes that can also be exploited.

These HTTPS implementation problems pose a great security risk to users. Different kinds of online attackers can exploit this situation to intercept and monitor user's communication for malicious purposes. There have already been several instances where these HTTPS issues have been used to compromise the security of websites [5].

Therefore, the aim of this study is to analyze the security of HTTPS implementations in government web-based systems in Tanzania. Similar security related studies have been done in other African countries. However, as studies have shown [28], ICT related issues are highly contextual in developing countries. The findings of this study will help to determine the existence of potential security and privacy vulnerabilities that can exploited. This will also assist in finding solutions that will establish a more secure and trustworthy environment.

2. RELATED WORK
HTTPS is one of the main security standards powering the Internet today [9]. According to Google's transparency report,

a large percentage of websites have been switching to HTTPS to encrypt their communication and authenticate their identity [14]. It involves the process of purchasing HTTPS certificates from certificate authorities and configuring webservers to support the HTTPS protocol. It also involves the configuration of other HTTPS parameters such as Perfect Forward Secrecy (PFS), and HTTP Strict Transport Security (HSTS) to ensure a complete and secure implementation.

Given the number of issues that need to be considered for secure implementation of HTTPS, many studies that focus on different aspects of HTTPS implementation have been done. For instance, several studies [3, 12, 37] performed security analysis on the HTTPS certificate ecosystem to determine different configuration problems that may pose security risks. They scanned over the whole Internet and found misconfigured trust relationships between certificate chains that can be exploited. Other studies such as [15], [32] have also analysed the aspect of HTTPS certificate validation and revocation and found several issues that can allow attackers to interfere with the security provided by HTTPS.

Similarly, several studies have been conducted to assess the security of government web-based systems. Zhao et al. [38] analyzed the security of government state websites in US to determine potential risks to users. The study used a sample of 51 official government websites to collect data over a period of two months. Using a combination of information from security audit and vulnerability assessment tools, the study found that 61\% of government websites had open ports that could allow cyber attackers malicious access to servers.

Akgul [2] evaluated the government websites in Turkey to determine security vulnerabilities against common web application attacks such as Cross Site Scripting (XSS) and SQL Injection (SQLi). A sample of 61 government websites selected from the Turkish government portal were scanned for 5 days in January 2016. The study found that security was given a low priority during implementation of government websites. As a result, 6\% of websites were found with critical web application vulnerabilities that could be easily exploited.

In developing countries, the situation is much worse. Awoleye [4] assessed the security of Nigerian government websites to enhance security of government websites. Over an interval of two years, the study scanned 64 randomly selected websites of government agencies and parastatals in Nigeria. The study found a significant number of government websites suffered from serious web application vulnerabilities such as XSS and SQLi.

Similarly, Bissyandé et al. [7] reviewed the security vulnerabilities of government websites in Burkina Faso to identify potential security holes that existed. Using automated tools, the study scanned 42 websites looking for well-known web application vulnerabilities. It found that about 50\% of government websites were using old versions of content management systems (CMS) with a significant number of security vulnerabilities.

These studies and many others show that a lot of effort has gone into analyzing the security of government web-based systems in various countries. However, minimal attention has been given on how HTTPS has been implemented and configured in these websites. With the increase in number of HTTPS vulnerabilities and attacks, users of government web-based systems are in great risks against security and privacy problems. Therefore, there is crucial need to perform security analysis of HTTPS implementations in government web-based systems to protect users and build trust that promotes the use of these systems.

3. METHODOLOGY

3.1 Sampling Process

The selected web-based systems for this study were sampled on a convenience basis. The focus was on web systems of major government organs, departments, institutions, and agencies. First, the selection process targeted public web-based systems of every ministry in the executive branch of the Tanzanian government. As of 2017, there are 20 ministries in the government of Tanzania. Each ministry has a web system that was included in the sample. Moreover, each ministry has several public institutions, departments, and organizations that are under its administration. Each of these had a web system that was also selected in the sample.

Other web-based systems of various public institutions and agencies that are listed in the Tanzanian Government Portal [19] were also included. The portal provides a comprehensive list of public web-based systems and services offered by entities that belong to the Tanzanian government. These were also included in the sample. In addition, a Google search of top public web systems with the go.tz domain was also conducted. Extra web systems not in the sample were found and added. As a result, a total of 74 web-based systems were sampled as shown in Table 1. Data collection was conducted in November 2017.

Table 1. A sample of government web-based systems selected for analysis

No	Government Entity	Domain
1	President's Office, Public Service Management and Good Governance	utumishi.go.tz
2	Architects and Quantity Surveyors Registration Board	aqrb.go.tz
3	Bank of Tanzania	bot.go.tz
4	Dar es Salaam Water and Sewerage Corporation	dawasco.go.tz
5	Dar es Salaam Rapid Transit Agency	dart.go.tz
6	Eastern Africa Statistical Training Centre	eastc.ac.tz
7	Energy and Water Utilities Regulatory Authority	ewura.go.tz
8	Higher Education Students Loan Board	heslb.go.tz
9	Higher Education Students Online Loan Application and Management System	olas.heslb.go.tz
10	Ministry of Agriculture	kilimo.go.tz
11	Ministry of Constitution and Legal Affairs	sheria.go.tz
12	Ministry of Defense and National Service	modans.go.tz
13	Ministry of Education, Science and Technology	moe.go.tz
14	Ministry of Finance and Planning	mof.go.tz

15	Ministry of Finance Planning Commission	mipango.go.tz
16	Ministry of Foreign Affairs and International Co-operation	foreign.go.tz
17	Ministry of Health, Community Development, Gender, Elderly and Children	mcdgc.go.tz
18	Ministry of Home Affairs	moha.go.tz
19	Ministry of Industry and Trade	mit.go.tz
20	Ministry of Information, Culture, Arts and Sports	habari.go.tz
21	Ministry of Lands, Housing and Human Settlements Developments	lands.go.tz
22	Ministry of Livestock and Fisheries	mifugouvuvi.go.tz
23	Ministry of Natural Resources and Tourism	mnrt.go.tz
24	Ministry of Water and Irrigation	maji.go.tz
25	Ministry of Works, Transport and Communication	mwtc.go.tz
26	National Addressing and Postcode System	address.go.tz
27	National College of Tourism	tourismcollege.go.tz
28	National Council for Technical Education	nacte.go.tz
29	National Defense College	ndctz.go.tz
30	National Development Corporation	ndc.go.tz
31	National Social Security Fund	nssf.or.tz
32	National Identification Authority	nida.go.tz
33	Parliament of Tanzania	parliament.go.tz
34	President's Office, Regional Administration and Local Government	tamisemi.go.tz
35	Prime Minister's Office	pmo.go.tz
36	Public Procurement Regulatory Authority	ppra.go.tz
37	Registration, Insolvency and Trusteeship Agency	rita.go.tz
38	Road Accident Information System	rais.mow.go.tz
39	Social Security Regulatory Authority	ssra.go.tz
40	Surface and Marine Transport Regulatory Authority	sumatra.go.tz
41	Tanzania Airports Authority	taa.go.tz
42	Tanzania Broadcasting Corporation	tbc.go.tz
43	Tanzania Buildings Authority	tba.go.tz
44	Tanzania Bureau of Standards	tbs.go.tz
45	Tanzania Commission for Aids	tacaids.go.tz
46	Tanzania Commission for Universities	tcu.go.tz
47	Tanzania Communications Regulatory Authority	tcra.go.tz
48	Tanzania Computer Emergency Response Team	tzcert.go.tz
49	Tanzania Electric Supply Company Limited	tanesco.co.tz
50	Tanzania Financial Intelligence Unit	fiu.go.tz
51	Tanzania Fire and Rescue Force	frf.go.tz
52	Tanzania Government Portal	tanzania.go.tz
53	Tanzania Immigration Department	immigration.go.tz
54	Tanzania Institute of Technology	tie.go.tz
55	Tanzania Insurance Regulatory Authority	tira.go.tz
56	Tanzania Meteorological Agency	meteo.go.tz
57	Tanzania National Bureau of Statistics	nbs.go.tz
58	Tanzania National Parks	tanzaniaparks.go.tz
59	Tanzania Online Mining Portal	portal.mem.go.tz
60	Tanzania Police Force	policeforce.go.tz
61	Tanzania Ports Authority e-Payment System	tpapayments.com
62	Tanzania Prevention and Combating of Corruption Bureau	pccb.go.tz
63	Tanzania Public Service College	tpsc.go.tz
64	Tanzania Railways Limited	trl.co.tz
65	Tanzania Revenue Authority	tra.go.tz
66	Tanzania Revenue Authority Gateway	gateway.tra.go.tz
67	Tanzania Rural Energy Agency	rea.go.tz
68	Tanzania Tourist Board	tanzaniatouristboard.go.tz
69	Vice President's Office	vpo.go.tz
70	Tanzania Wildlife Management Authority	tawa.go.tz
71	The Judiciary of Tanzania	judiciary.go.tz
72	The National Examinations Council of Tanzania	necta.go.tz
73	The Public Procurement Regulatory Authority	ppra.go.tz
74	Vocational Education and Training	veta.go.tz

	Authority	

3.2 Evaluation Process

The process of implementing HTTPS on a web-based system involves configuring a variety of security parameters (See Table 2). Each of these parameters must be configured properly to guarantee an overall secure implementation. Several organizations that focus on web security have provided a series of security checks and best practices for implementing HTTPS. The Open Web Application Security Project (OWASP) [30] which is an international organization focused on improving software security has published a Transport Layer Protection Cheat Sheet [22]. The cheat sheet provides a simple list to follow for securely implementing HTTPS in a web application. Similarly, Qualys, under their SSL Labs research efforts have also published HTTPS security guidelines to help with secure assessment and configuration of HTTPS in web-based systems [24].

Table 2. Security parameters to be configured in HTTPS implementations

Parameter	Description
Perfect Forward Secrecy (PFS)	Ensures currently captured communication is protected from future compromise of private secured keys
Server Cipher Preference	HTTPS Server specifies the order cryptographic protocols it prefers to prevent downgrade attacks
Online Certificate Status Protocol (OCSP)	Protocol to verify the revocation status of a provided HTTPS certificate. Guarantees the validity of a certificate
HTTP Strict Transport Security (HSTS)	Parameter that forces clients to use HTTPS only. Protects web systems against downgrade attacks
HTTPS Certificate Parameters	All fields of information included in a certificate such as organization name, domain, certificate authority and expiration date
DNS Certification Authority Authorization (DNS CAA)	Specifies the certificate authority (CA) that can issue certificate for the domain. Prevents impersonation from unauthorized CAs
Certificate Revocation List (CRL)	List of HTTPS certificates revoked before expiration date. Verifies the validity and trustworthiness of a certificate

There are also several known security vulnerabilities that exist in HTTPS implementations. Table 3 shows these vulnerabilities with a brief description of their impact. These vulnerabilities have unique signatures that can be easily identified and evaluated during analysis.

Table 3. Known vulnerabilities in HTTPS implementations

Vulnerability	Description
Browser exploit against SSL/TLS (BEAST)	Attack against TLSv1 that exploit weaknesses in cipher block chaining (CBC) mode of the protocol. Can enable man-in-the-middle attacks where attacks can get access to data exchanged between client and server. Discovered in 2011.
Browser Reconnaissance and Exfiltration via Adaptive Compression of Hypertext (BREACH)	Security vulnerability that exploits the compression algorithm used in HTTPS. Can be used to extract secured information in communication such as login details. Disclosed in 2013.
Change Cipher Spec (CCS)	Vulnerability that allows attackers to inject change of cipher spec messages in HTTPS communications. Can allow man-in-the-middle attacks. Disclosed in 2014
Compression Ratio Info-leak Made Easy (CRIME)	Vulnerability exploited from weaknesses in compression algorithms used in HTTPS. Can allow extraction of secret authentication tokens leading to session hijacking.
Decrypting RSA with Obsolete and Weakened Encryption (DROWN)	Vulnerability that affect HTTPS implementations that still have support of the old version of SSL (SSLv2). Allows attackers to break the encryption and steal sensitive data.
LUCKY13	Attack against HTTPS implementation that use the CBC mode of encryption. Does not pose significant danger but can potentially lead to information leak
Padding Oracle on Downgraded Legacy Encryption (POODLE)	Vulnerability that can allow interception of secured HTTPS communication. Exploits implementations that fall back to old insecure SSLv3 protocol

Based on these guidelines and vulnerabilities, researchers have developed automated tools to determine the security status of HTTPS implementations. These tools fall under two main categories: online and offline. Online tools include SSL Checker [25], SSL Server Test [23], and Observatory [18]. They are operated online through a web-based interface. They also provide their results through the same web interface. Some allow these results to be easily exported for further analysis. Offline tools include SSLScan [34], SSLyze [10], and testssl [36]. These tools can be operated locally, typically through a command line interface. They allow results to be displayed on the command line for further analysis. To allow for easy analysis and comparison, offline tools were preferred for this study. Among the offline tools, the testssl tool was chosen as it has a unique feature that allows multiple web systems to be scanned concurrently.

Therefore, using the testssl tool, a total of 74 websites were checked and evaluated for different HTTPS configuration parameters. The tool tested each parameter to see if it was configured according to the recommendations in the guidelines. The tool also tested the existence of known HTTPS vulnerabilities. A detailed report for each web system containing quantitative and qualitative data was then generated. The quantitative data was based on how many checks and tests each web system passed while the qualitative data described the nature of security failures that were discovered. The results are explained next.

4. RESULTS

The results from scanning each web system were collected, aggregated, and analyzed to identify problem categories and trends.

4.1 HTTPS Support

Before the security of the HTTPS implementation could be analyzed, the web systems were first checked to see if they supported HTTPS. The results showed that out of 74 scanned web-based systems, 43 (58\%) had implemented HTTPS while 31 (42\%) did not implement HTTPS. For those 58\% web-based systems that had implemented HTTPS, an analysis was performed to determine how many supported each version of the protocol. It was found that all 43 web systems implemented TLSv1.0 version of HTTPS. Most of the systems (91\%) had also implemented TLS versions 1.1 and 1.2. Notably, only one of the systems (i.e. fiu.go.tz) was offering SSLv2 which is the oldest version of the protocol. In addition, none of the web systems implemented TLSv1.3 which is the most recent version of HTTPS. Figure 1 shows the distribution of these versions.

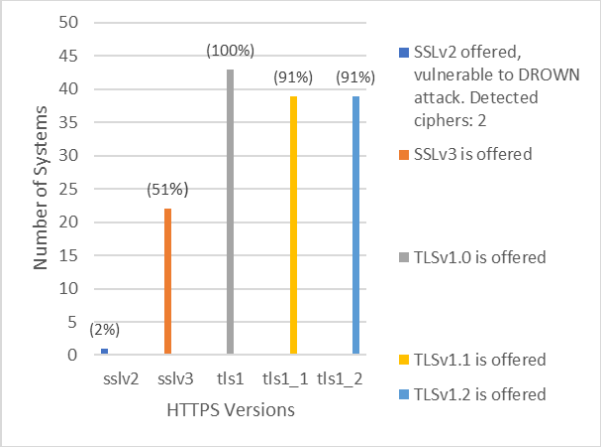

Fig 1: Distribution of HTTPS Versions

4.2 HTTPS Security Problems Severity

For each scanned web system, the testssl tool looked for and identified different HTTPS security problems according to the criteria specified in Table 2 and Table 3. The results show that every web system that had implemented HTTPS had multiple number of security problems that need to be addressed. In general, the number of issues found on each web system ranged from 8 in the National Bureau of Statistics system (nbs.go.tz) to 31 in the Tanzanian Institute of Education system (tie.go.tz). On average the number of issues that were found in each system was 16.

These issues were then analyzed, sorted and grouped into three main security priority levels. Table 4 describes these priority levels.

Table 4. Priority Levels of HTTPS Implementation Problems

PRIORITY	LOW	MEDIUM	HIGH
DESCRIPTION	These are non-critical HTTPS configurations that have not been implemented or offered	These are HTTPS misconfiguration issues that may lead to security compromise	These are critical HTTPS configuration issues that can be easily exploited.
ACTION	It is good security practice if they are implemented or offered	It is recommended that they are fixed	It is highly recommended that they are fixed immediately.

Results show that out of all issues, there were more HIGH priority security issues compared to MEDIUM and LOW. Figure 2 shows the percentage division of security priority levels.

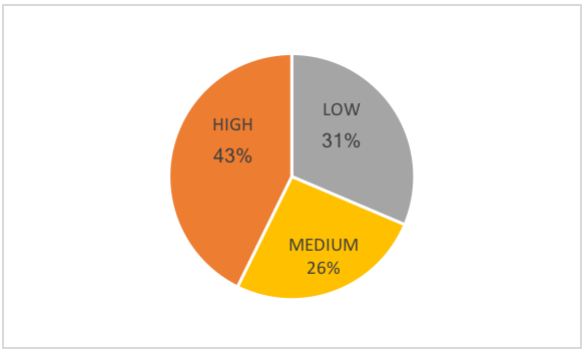

Fig 2: Proportion of HTTPS Problems in terms of priority

Every web system that was scanned had the whole range of issues from LOW to HIGH priority. Each system had a minimum of two HIGH priority issues that need to be fixed immediately. Notably, tie.go.tz which had the highest number of issues found, also had the highest number of HIGH priority issues. Table 5 shows the breakdown of number of problems in each category for the Top 15 systems with the highest number of issues.

Table 5. Number of HTTPS Implementation problems found in each category for the Top 15 systems

PRIORITY	LOW	MEDIUM	HIGH
tie.go.tz (197.149.176.219)	18	3	10
utumishi.go.tz (216.198.246.99)	14	4	6
pccb.go.tz (216.198.246.103)	13	3	6
veta.go.tz (216.198.246.105)	13	4	6
meteo.go.tz (154.118.231.8)	12	5	3
ssra.go.tz (197.149.176.23)	12	5	3
tawa.go.tz (154.118.230.25)	12	5	3
parliament.go.tz (197.149.176.23)	12	5	3
mipango.go.tz (216.198.246.100)	12	3	5
nida.go.tz (41.59.254.121)	12	4	6

address.go.tz (41.188.170.9)	12	4	8
bot.go.tz (196.46.101.39)	10	6	4
fiu.go.tz (41.221.50.22)	10	4	7
tpapayments.com (196.43.221.17)	9	5	8
tanzania.go.tz (197.149.176.23)	7	6	3

4.3 HTTPS Security Configuration Problems

Looking more closely into the specific nature of these issues, results show that one of the most common HIGH priority security issue found was the vulnerability to the BEAST attack [11]. This affected all systems. Likewise, the lack of HTTP Strict Transport Security (HSTS) [16] was also another common occurring issue. All systems had HSTS disabled. This means all systems were vulnerable to protocol downgrade [8] and cookie hijacking [31] attacks. Other common occurring security issues include support for old vulnerable protocols such as Triple DES and SSLv3, support for 128-bit weak ciphers such as SEED and IDEA, and other misconfiguration such as lack of cipher order preference and certificate mismatch. Table 6 shows the top 15 most common HIGH priority security issues that were found and their number of occurrences.

Table 6. Top 15 Common Security Configuration Problems

Configuration Problem	No
No support for HTTP Strict Transport Security	43
Vulnerability to BEAST	43
Potentially vulnerable to LUCKY13	41
Vulnerability to SWEET32	38
Weak Cipher Triple DES Ciphers (Medium) offered	38
No security headers detected	33
Weak 128 Bit ciphers (SEED, IDEA, RC4) offered	30
Common prime 'RFC3526/Oakley Group 14' detected	29
Server does NOT set a cipher order	28
Old protocol SSLv3 is offered	22
Vulnerability to POODLE (uses SSLv3+CBC)	22
OCSP stapling not offered	18
Certificate does not match supplied URI (same w/o SNI)	15
Support vulnerable RC4	14
SubjectAltName (SAN) not provided	10

Other serious but rare issues were also found. Ten systems had implementations with self-signed certificates. This included systems under the following domains: meteo.go.tz, mipango.go.tz, nida.go.tz, parliament.go.tz, pccb.go.tz, ssra.go.tz, tawa.go.tz, tie.go.tz, utumishi.go.tz, and veta.go.tz. Several other systems had certificates that had expired. This included systems from bot.go.tz, tanzania.go.tz and tira.go.tz. Particularly, there were some systems such as meteo.go.tz, parliament.go.tz, and tie.go.tz that had both self-signed and expired certificates. In addition, there were other systems that had untrustworthy HTTP certificates chains. This means these systems are offering certificates that cannot be trusted by user's browsers. Table 7 shows the most critical security configuration issues that were found.

Table 7. Most Critical HTTPS issues found

Problem	No
All certificate trust checks failed: (chain incomplete).	5
All certificate trust checks failed: (expired).	7
Vulnerable to CCS	2
Self-Signed Certificates	10
LOW: 64 Bit + DES encryption (w/o export) offered	2
SSLv2 offered, vulnerable to DROWN attack	1
TLSv1.1 is not offered, and downgraded to a weaker protocol	4
TLSv1.3: connection failed rather than downgrading to TLSv1.2	1

4.4 Vulnerabilities

The HTTPS implementation on each system was also tested against a known list of HTTPS vulnerabilities as specified in section 3.2. Each of these vulnerabilities has a specific unique signature which the testssl identifies during the scanning process. Results show that all systems were vulnerable to the BEAST attack [11]. Therefore, almost all systems were also potentially vulnerable to the LUCKY13 attack [17]. Furthermore, many systems were vulnerable to POODLE [21], SWEET32 [6], and RC4 [29]. Figure 3 shows the number of systems affected by each vulnerability.

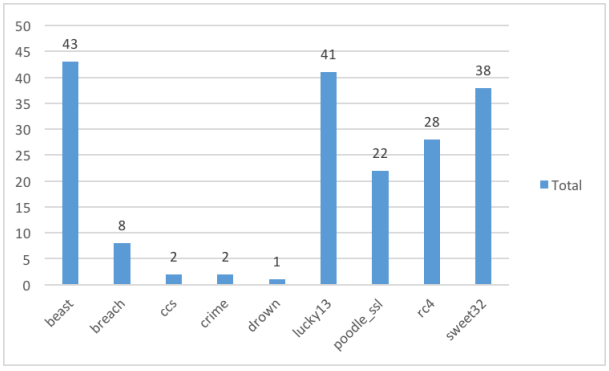

Fig 3: Distribution of total number of Government Web Systems with each HTTPS Vulnerability

Each system was also analyzed to determine critical number of vulnerabilities that need immediate attention. Results show that all systems had implementations that were affected by multiple vulnerabilities at the same time. Systems under the Tanzania Institute of Technology (tie.go.tz) and National Addressing and Postcode System (address.go.tz) domains had the greatest number of vulnerabilities present. Both had 7 out of the 9 critical vulnerabilities present. Only nbs.go.tz, rita.go.tz, and olas.heslb.go.tz had 2 vulnerabilities which was the least number of vulnerabilities present. Table 8 shows top

systems that had 6 or more vulnerabilities present at the same time.

Table 8. Top web systems with the greatest number of HTTPS vulnerabilities

System	beast	breach	ccs	crime	drown
tie.go.tz	Y	-	Y	Y	-
address.go.tz	Y	-	Y	Y	-
utumishi.go.tz	Y	Y	-	-	-
tawa.go.tz	Y	Y	-	-	-
ppra.go.tz	Y	-	-	-	-
mcdgc.go.tz	Y	Y	-	-	-
veta.go.tz	Y	Y	-	-	-
fiu.go.tz	Y	-	-	-	Y

5. DISCUSSION

The Tanzanian government through its e-government agency (eGA) has been consolidating and increasing its efforts in using ICT systems to provide public services. These efforts have allowed the government to increase the reach of its services. More people can obtain important public information and perform functions at the own convenience wherever they are. Consequently, this has allowed the government to reduce its operation costs, increase transparency, and reduce bureaucracy. One of the key technologies that has facilitated this has been the increased availability of Internet services. According to the Tanzania Communications Regulatory Authority (TCRA) quarterly communication statistics of June 2017 [33], the estimated number of Internet users is about 20 million. This translates to 40\% penetration. Therefore, many Tanzanian citizens interact with various systems over the web.

One of the key aspects for the successful and effective use of public web-based systems is security. The presence of security in these systems not only protects people from cyber-attacks, but also builds trust in using them. According to Abu-Shanab [1], privacy and security play a critical role towards the perception of trust in government services. Visible and effective security measures ensure people that their data and operations are well protected. This in turn also influences the adoption rate and willingness to continue to use these systems. Many previous studies have looked at different security aspects of government web systems. However, little attention has been paid on the security of HTTPS implementations.

The goal of this study was to perform a security analysis of HTTPS implementations in government web systems. The main reason is that HTTPS is one of the most visible forms of security implemented in web-based systems. It encrypts the communication over the Internet and allows people to interact with government web systems securely. However, it is also one of the security measures that once implemented tends to be forgotten. Administrators of these systems believe that once HTTPS certificates have been purchased and installed, security has been taken care of. However, history has proven otherwise. Many other configurations and measures must be considered to ensure a secure implementation.

Therefore, the goal of this study was achieved by taking a sample of 74 Tanzanian government web domains and scanning them using the testssh tool. The tool checked for different HTTPS security parameters such as default server configurations, certificate parameters, support for different versions of the protocol, and protection against well-known HTTPS vulnerabilities. The detailed report from the tool was then analyzed to evaluate the security status of these HTTPS implementations. The analysis revealed that many Tanzanian government web systems lacked support for HTTPS, suffered from high risk HTTPS vulnerabilities, supported old and vulnerable protocols, had a lot of server misconfigurations, and offered invalid HTTPS certificates.

The first aspect that was considered was the proportion of government web systems that supported HTTPS. This study found that only 58\% of the systems (43 out of 74) had implemented HTTPS in their web interfaces. This means a significant portion of Tanzanian government systems do not protect their communication over the web. This puts citizens in significant risk. Many of these systems that have not implemented HTTPS, require users to login to perform different operations. Some examples include the Universities Information Management System (uims.tcu.go.tz), the Foreign Award Assessment System (faas.tcu.go.tz), the Student's Admission Verification System (http://nacte.go.tz/), and the Abnormal Load Permit System (epermit.mow.go.tz/). Users of these systems are under risk of disclosing sensitive login information. In addition, even for systems that only provide information, HTTPS still protects their web interfaces from being forged or modified.

Looking at the severity of existing HTTPS security problems, approximately 43\% of the systems with HTTPS implementations were found to have high priority security issues. These are issues that require immediate attention and fixing. They put the concerned systems in potential high risk of compromise. They include issues such as support for old protocols like SSLv3 proven to be vulnerable [26], support for weak 128bit ciphers like SEED and IDEA which are insecure [35], lack server cipher preference which can allow downgrade attacks [8], and mismatched certificates which defeat the whole purpose of HTTPS certificates. These are all issues that are commonly overlooked during HTTPS implementations. However, they have significant impact on the overall security status of the web systems.

Focusing on HTTPS certificates themselves, the study also found many issues. About 23\% (10 out of 43) of the web systems were offering self-signed certificates. This is not the standard practice for publicly available systems, especially government ones. This type of certificates causes huge security warnings in user's browsers and decreases their trust in the concerned systems. Other issues such as expired and broken certificates were also found. All these are basic and standard things that every HTTPS implementation should provide at the minimum. The existence of these issues indicates a lack of proper security attention for many of these systems and could be a signal for bigger problems.

Finally, the study also found that a signification proportion of systems had HTTPS implementations with known vulnerabilities. All systems were vulnerable to the BEAST attack [11]. This is an attack that enables an intruder to silently decrypt secure communication between a user and the web-based server. It affects old versions and implementations of HTTPS. More than 90\% of the systems were also vulnerable to the LUCKY13 and SWEET32 attacks. Specifically, every single system had more than one vulnerability. Despite these vulnerabilities being well-known with fixes available, many of these government systems are yet to be patched putting them in potential high security risk. Since these issues are well-known, they can be easily

compromised by attackers. Security fixes need to be applied as soon as possible.

6. FUTURE RESEARCH

This study focused on analyzing the security status of government web systems HTTPS implementations by using an automated tool that observes HTTPS responses received from the systems. Despite the significant results obtained, there are some limitations with this approach. Since the analysis of the tool depends on the HTTPS responses it receives remotely, any interference with this process might lead to erroneous conclusions. For example, firewalls or other security systems might modify or filter some of the parameters in the responses. In addition, this approach might not identify other HTTPS issues such as private key protection that do not reveal themselves in the responses. Future studies should focus on investigating local server to increase the scope of the analysis.

HTTPS is just one security aspect of government web-based systems. There are many other aspects that contribute to the overall security of a system such user authentication, web application vulnerabilities, and database security. These are more backend issues that require greater access and more cooperation from the respective institutions. Future research should also try to find means to do security analysis of these other important aspects.

7. CONCLUSION

In Tanzania, as more government services become available over the web in the form of web-based systems, security of these systems also becomes a critical component that needs to be addressed. Lack of enough security puts citizens under cyber risks and decreases the level of trust on the systems. With HTTPS being one of the most common and visible security aspects of web-based systems, it is important that it is implemented correctly and securely. However, this study has shown that many important government systems in Tanzania have yet to implement HTTPS in their web interfaces. In systems where it has been implemented, many issues were found including misconfigurations, certificate problems, support of old and weak protocols, and vulnerability to well-known security attacks.

It is crucial that government web-based systems that are accessible over the Internet are also secure and safe to use. Unlike private systems which users can opt not to use or find alternatives, public systems do not have that luxury. People must use them to get important information and services. If these systems are not secure, everyone will be at risk. It is important for the government to protect its people and build confidence in its operational abilities.

8. REFERENCES

[1] Emad Abu-Shanab. Antecedents of trust in e-government services: an empirical test in Jordan. Transforming Government: People, Process and Policy, 8(4):480–499, oct 2014.

[2] Yakup Akgul. Web Site Accessibility, Quality and Vulnerability Assessment: a Survey of Government Web Sites in the Turkish Republic. Journal of Information Systems Engineering & Management, 1(4):1–13, 2016.

[3] Axel Arnbak, Hadi Asghari, Michel Van Eeten, and Nico Van Eijk. Security collapse in the HTTPS market. Communications of the ACM, 57(10):47–55, 2014.

[4] Olusesan M. Awoleye, Blessing Ojuloge, and Mathew O. Ilori. Web application vulnerability assessment and policy direction towards a secure smart government. Government Information Quarterly, 31(S1): S118–S125, 2014.

[5] Ionu-Daniel Barbu and Ioan Bacivarov. Heartbleed The Vulnerability That Changed the Internet. International Journal of Information Security and Cybercrime, 3(2):49–60, dec 2014.

[6] Karthikeyan Bhargavan and Leurent Gaëtan. Sweet32: Birthday attacks on 64-bit block ciphers in TLS and OpenVPN, 2016.

[7] Tegawendé F. Bissyandé, Jonathan Ouoba, Daouda Ahmat, Fréderic Ouédraogo, Cedric Béré, Moustapha Bikienga, Abdoulaye Sere, Mesmin Dandjinou, and Oumarou Sié. Vulnerabilities of government websites in a developing country: The case of Burkina Zaso. In Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST, volume 171, pages 123–135, 2016.

[8] Jeremy Clark and Paul C. Van Oorschot. SoK: SSL and HTTPS: Revisiting past challenges and evaluating certificate trust model enhancements. In Proceedings - IEEE Symposium on Security and Privacy, pages 511–525, 2013.

[9] T. Dierks and E. Rescorla. RFC 5246 - The transport layer security (TLS) protocol - Version 1.2. In Network Working Group, IETF, pages 1–105, 2008.

[10] Alban Diquet. SSLyze - Fast and powerful SSL/TLS server scanning library, 2017.

[11] Thai Duong. BEAST, 2011.

[12] Zakir Durumeric and James Kasten. Analysis of the HTTPS certificate ecosystem. IMC '13 Proceedings of the 2013 conference on Internet measurement conference, pages 291–304, 2013.

[13] Yoel Gluck, Neal Harris, Angelo Angel, and Prado. BREACH: REVIVING THE CRIME ATTACK. 2013.

[14] Google. Transparency Report, 2017.

[15] Nils Gruschka, Luigi Lo Iacono, and Christoph Sorge. Analysis of the current state in website certificate validation. Security and Communication Networks, 7(5):865–877, 2014.

[16] J. Hodges, C. Jackson, and A. Barth. HTTP Strict Transport Security (HSTS). Technical report, nov 2012.

[17] G. Irazoqui, M.S. Inci, T. Eisenbarth, and B. Sunar. Lucky 13 strikes back. ASIACCS 2015 - Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security, pages 85–96, 2015.

[18] April King. Observatory by Mozilla, 2016.

[19] Arts Ministry of Information, Culture and Sports. Tanzania Government Portal: Welcome, 2017.

[20] Ministry of Works Transport and Communication. National Information and Communications Technology Policy, 2016.

[21] Bodo Möller, Thai Duong, and Krzysztof Kotowicz Google. This POODLE Bites: Exploiting the SSL 3.0 Fallback Security Advisory. 2014.

[22] OWASP. Transport Layer Protection Cheat Sheet - OWASP, 2017.

[23] Qualys. SSL Server Test (Powered by Qualys SSL Labs), 2017.

[24] Qualys. SSL/TLS Deployment Best Practices, 2017.

[25] Rapid Web Services. SSL Checker- Check SSL Certificate of Website with Free SSL Checker Tool, 2017.

[26] Ivan Ristic. SSL 3 is dead, killed by the POODLE attack — Qualys Blog, 2014.

[27] Juliano Rizzo and Thai Duong. Crime: Compression ratio info-leak made easy. In ekoparty Security Conference, 2012.

[28] Oystein Sæbø. E-government in tanzania: Current status and future challenges. In Electronic Government, pages 198–209. Springer Berlin Heidelberg, 2012.

[29] Santanu Sarkar, Sourav Sen Gupta, Goutam Paul, and Subhamoy Maitra. Proving TLS-attack related open biases of RC4. Designs, Codes, and Cryptography, 7 (1):231–253, 2015.

[30] William Schmidt. Open web application security project. Open Web Application Security Project, pages Vulnerability Table, paragraph 2, 2009.

[31] Suphannee Sivakorn, Iasonas Polakis, and Angelos D. Keromytis. The Cracked Cookie Jar: HTTP Cookie Hijacking and the Exposure of Private Information. In Proceedings - 2016 IEEE Symposium on Security and Privacy, SP 2016, pages 724–742, 2016.

[32] Pawel Szalachowski, Laurent Chuat, Taeho Lee, and Adrian Perrig. RITM: Revocation in the Middle. apr 2016.

[33] Tanzania Communications Regulatory Authority. Quarterly Communications Statistics Report - June 2017, 2017.

[34] Ian Ventura-Whiting. SSLScan - Fast SSL Scanner download — SourceForge.net, 2013.

[35] Xiaoyun Wang and Hongbo Yu. How to Break MD5 and Other Hash Functions. pages 19–35. 2005.

[36] Dirk Wetter. Testing TLS/SSL encryption, 2017.

[37] Liang Zhang, David Choffnes, Dave Levin, Tudor Dumitras, Alan Mislove, Aaron Schulman, and Christo Wilson. Analysis of SSL certificate reissues and revocations in the wake of heartbleed. In Proceedings of the 2014 Conference on Internet Measurement Conference - IMC '14, pages 489–502, 2014.

[38] Jensen J. Zhao, Sherry Y. Zhao, and Sherry Y. Zhao. Opportunities and threats: A security assessment of state e-government websites. Government Information Quarterly, 27(1):49–56, 2010.