Secure Encrypted Connection on Georgian Website

Giorgi Akhalaia[1], Maksim Iavich[2], Giorgi Iashvili[2], Dmytro Prysiazhnyy[3], and Tetiana Smirnova[4]

[1] Georgian Technical University, 77 Kostava str., Tbilisi, 0160, Georgia
[2] Caucasus University, 1 Paata Saakadze str., Tbilisi, 0102, Georgia
[3] Vinnytsia National Technical University, 95 Khmelnytske ave., Vinnytsia, 21021, Ukraine
[4] National Aviation University, 1 Liubomyra Huzara ave., Kyiv, 03058, Ukraine

Abstract

We make an effort and spend loads of time trying to secure IT infrastructure and services. We hide the entire network segment behind firewalls, DMZs, and other security mechanisms to protect from data breaches and interception. But, one point remains—websites. They are open targets and are in the first line for attackers. Except for common types of web attacks, like a DoS, a misconfigured webpage is vulnerable for every user connected to it. This article is about how securely Georgian websites are configured, generally concerning HSTS. Which is a powerful protection against MITM attacks. The study covers the main aspect of HSTS parameters, describes major problems in Georgia, and designs how they should be resolved. According to research, only 1% of Georgian websites are served under HSTS. Also, 39% of webpages are accessible via HTTP. The majority of them have HTTPS (HTTP with encryption and verification) support, but because of misconfiguration, users face critical security issues. In very populated cities, like Tbilisi, there are high availability of free wireless networks. This increases the risk of getting intruders and targets in the same network. Which itself doubles the probability of data breach, network sniffing, and so on. The level of user awareness is very low, so it is crucial to maintain web servers so securely, that minimize user-side vulnerabilities.

Keywords

HSTS, preload service, website security, HTTPS, encryption.

1. Introduction

Web services have become extremely popular over the last few years. Thus, increased web-related threats for both sides: clients and servers. Talking about web security usually leads users to think about HTTPS (HTTP with encryption and verification). However, redirection from HTTP to HTTPS does not guarantee secure communication between the user and the website [1–5]. There are techniques for downgrading HTTPS connection to HTTP, which is at higher risk of sniffing. Hence, there was a need for some security mechanism that would try to compensate for this vulnerability. In November of 2012, RFC proposed a standard, known as RFC 6797, which defined a security mechanism for websites—HTTP Strict Transport Security (HSTS) [6].

HSTS is a kind of security technology that protects web pages from HTTPS downgrade, also known as an "SSL strip attack." These are types of MITM (man-in-the-middle) attacks when a hacker stands between the user and web server and converts HTTPS connection to HTTP (Fig. 1). HSTS is a method used by webpages to announce that it must be accessed only via HTTPS. When the HSTS policy is declared by the website, browsers refuse every HTTP request (Fig. 2). Most modern web browsers

CPITS-2023-II: Cybersecurity Providing in Information and Telecommunication Systems, October 26, 2023, Kyiv, Ukraine
EMAIL: g.akhalaia@gmail.com (G. Akhalaia); miavich@cu.edu.ge (M. Iavich); giashvili@gmail.com (G. Iashvili); dimpris@gmail.com (D. Prysiazhnyi); t.smirn@gmail.com (T. Smirnova)
ORCID: 0000-0002-4194-2681 (G. Akhalaia); 0000-0002-3109-7971 (M. Iavich); 0000-0002-1855-2669 (G. Iashvili); 0009-0000-8327-3183 (D. Prysiazhnyi); 0000-0001-6896-0612 (T. Smirnova)

support the HSTS protocol. However, correctly configured HSTS protocol does not assure full protection from interception. In the case of a freshly installed OS or web browser, there is always a way to grab the connection. The first attempt to contact the website can be captured. Because, before the web browser gets the HSTS directive from a webpage, by default, it will try to connect via HTTP.

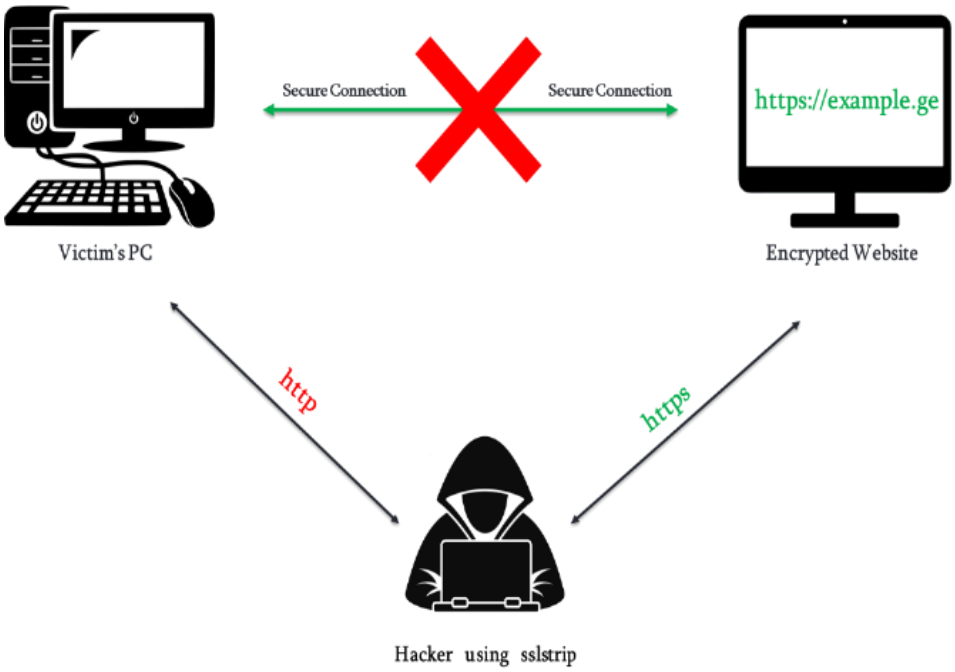

Figure 1: HTTP and HTTPS connection schemes

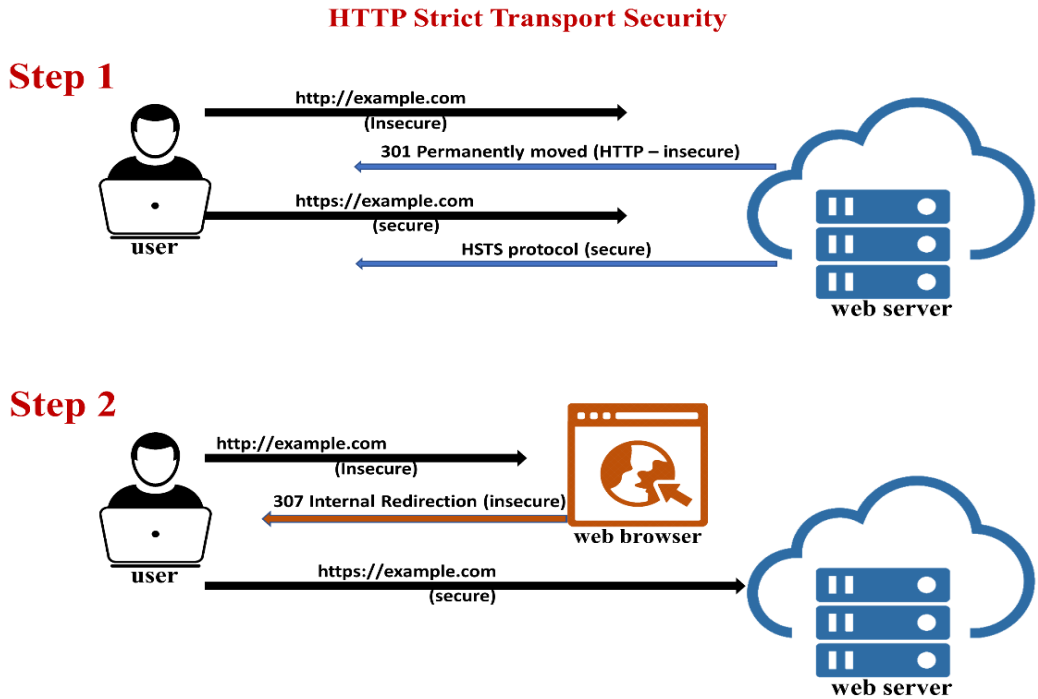

Figure 2: HTTP Strict Transport Security

To override that threat, Google has started the HSTS preload service. If the website is submitted for preload service, it will be hardcoded in web browsers and will not be accessible via HTTP. Before a connection is established between the computer and the webpage, HSTS-compatible web browsers check their internal HSTS list. If the website

is preloaded, the browser will automatically use the HTTPS protocol. Google HSTS preload service is free, but there are some security requirements for submission:

1. A valid Certificate.
2. Redirection from HTTP to HTTPS.
3. All subdomains must be served over HTTPS.
4. Proper configuration of HSTS Header.

Properly configured HSTS protocol does not mean total protection from MITM, but it will significantly secure client-server conversation.

2. Survey

An important part of this article is the security assessment of connections on Georgian websites. Hence, research was done on randomly selected Georgian websites. Hundreds of pages were tested on HSTS protocol (compatibility and/or eligibility) during the research. Websites were tested using the Google online platform for HSTS preload service status and eligibility checking [7]. During the survey web pages were checked in the following parameters:

- Online Transaction/Authorization availability on the website.
- HTTP / HTTPS access.
- HTTPS Auto Redirection.
- HSTS Header availability.
- If the webpage serves all subdomains over HTTPS.
- If the HSTS preload service is already active.
- If the webpage was eligible for HSTS preload service.

For classifying the Georgian market, the target group was divided by ownership (State or Private) and the following categories: Communication; Education; Banks and Finances, Healthcare and Lifestyle; National Centers and Agencies; TV companies; News; Oil and Petroleum; State Structures and Services; Entertainment; Online shopping; Utility payments and other bills; Bookmakers and other online games; Adult; Logistics.

The first security parameter was "Online Transaction/Authorization." This is very important because if the user makes a transaction or authorization on the webpage, it means personal, sensitive data is transmitted between the user and the web server, so this parameter increases the risk of eavesdropping or other MITM-type attacks [8–13]. 83% of checked websites have online transaction and/or authorization modules (T/A module). The majority of web pages without a T/A module (17%) are from informational categories. The second parameter was if the website was accessible using HTTP protocol. It seems quite unserious while in the age of cyberwar, and cyberterrorism we are discussing again the existence of HTTP conversation between web servers and users, but according to research, 39% of Georgian websites can be accessed using HTTP. If we correlate the first two parameters: websites with the T/A module and HTTP-accessible webpages we'll find, that 36% of the sites with the T/A module can be accessed via HTTP (Fig.3) [10, 12].

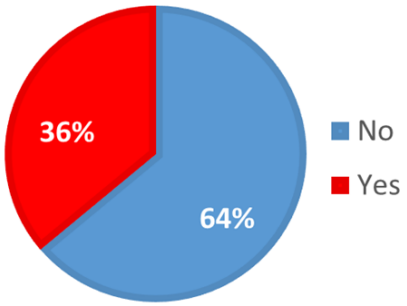

Figure 3: Sites access statistics

So, in 36% of cases, while the user makes authorization or transaction, sensitive data is transmitted as a clear text and can be stolen without the significant effort of a hacker. 37% of them are operated by state institutions or structures and 63% are private business operators (Fig. 4) [11, 14].

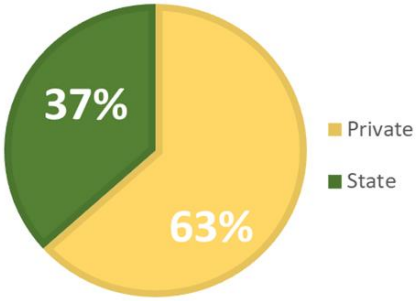

Figure 4: HTTP accessible statistics

We would like to mention, that according to an analysis, 47% of this category does not have HTTPS support. We mean, there is no way to connect these servers via HTTPS [13].

The next parameter was auto redirection from HTTP to HTTPS. 31% of webpages are not configured so, to redirect every HTTP request to HTTPS. Hence, even if the webpage has HTTPS support, there is always a chance to listen to traffic and conversation between the victim and server computer. If we analyze websites with T/A modules concerning HTTP auto redirection enabled webpages, we'll see more lack of security. 77% of this category does not make HTTP redirection (Fig. 5) [15].

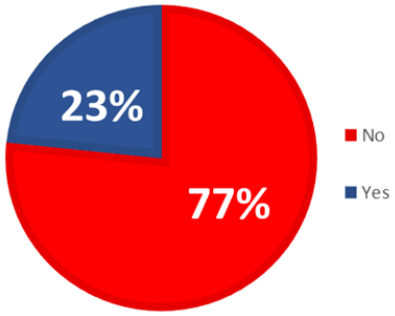

Figure 5: Site redirection statistics

Let's dig a bit more into the results. There is one important thing, in 30% of the webpages with T/A module connection established via HTTP, because of the misconfigured auto redirection directive (Fig. 6). To be clearer, in 30%, connections can be more secure but without automatic redirection, when user types "example.com" he/she establishes HTTP, nonsecure conversation (because browsers default port is 80, same as HTTP). So, it's an extra window for hackers to intercept conversations and listen, modify, redirect, or add packets.

As it was mentioned above, the main reason for this study was to check how properly Georgian webpages (as an example) are configured on the HSTS security mechanism and if they are eligible for HSTS preload service. Study shows that only 21% of websites have an HSTS directive in the header (Fig. 7) [13–15].

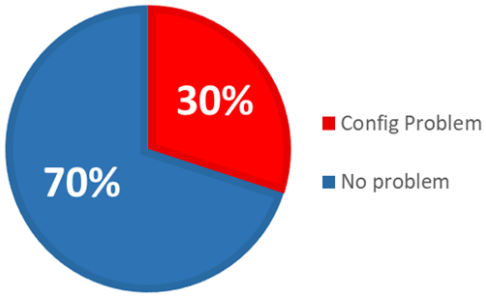

Figure 6: Site problems statistics

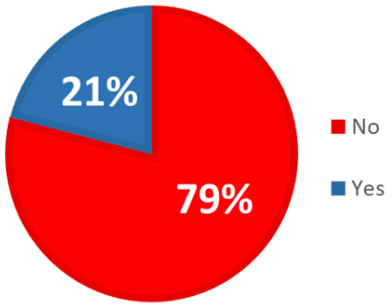

Figure 7: HSTS site statistics

If we check the target group without an HSTS header (79%) on HTTPS support, we'll get the following results: 73% of them can be accessed via HTTPS (Fig. 8). But, because of the missing HSTS directive, there is a huge window for SSL Strip attack. That's why the HSTS header takes so important place after HTTPS protocol in the security of web technologies [16].

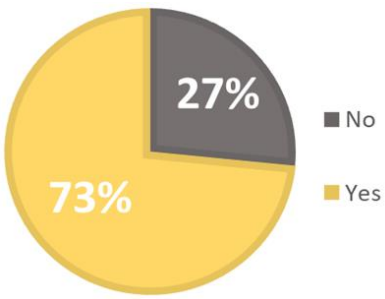

Figure 8: HSTS sites with HTTPS statistics

In common cases, developers take care only about domains and they are not attentive to subdomains. I mean, usually, only main domains are served by HTTPS protocol, and subdomains, or a major part of them, are left under HTTP. During the survey, there were cases when the main site was under HTTPS, but when I was entering the authorization page, which was on the subdomain, it was served only via HTTP [17–19].

So, it was a critical breach of confidentiality. In our cases 43% of a total of the target group serves all subdomains over HTTPS (Fig. 9). If we analyze this data concerning HTTPS-enabled web pages, we'll see, that in 28% cases of, the privacy of the user is violated (Fig. 10).

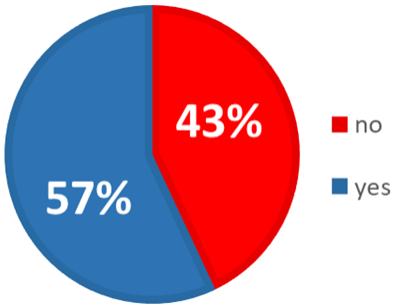

Figure 9: Subdomains statistics

The last two criteria were about HSTS preload service. The first one was if the website had already been preloaded and the second one was if they were eligible for Google preload service. Unfortunately, only 1% of Georgian webpages are preloaded and no one except this 1% was not eligible for preloading [20–22].

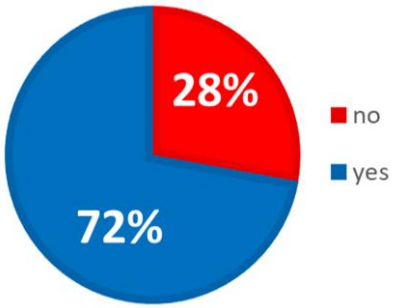

Figure 10: Subdomains over HTTPS statistics

According to the results, it is clear, that there should be done lots of work to improve the security level of Georgian Websites. Some brief explanations about checking security mechanisms and recommendations will be described in the following paragraphs [23–26].

3. Checking Websites on HSTS

As mentioned above, Google has started the HSTS preload service. So, they provide an online platform for checking if the website is correctly configured on HSTS security protocol and if it is eligible for preload service. There is a checklist of HSTS parameters which support is mandatory [18–21].

Before going deep into the HSTS details, let us see what the HSTS preload directive looks like.

The very first step before configuring the HSTS policy is to serve the site with valid certificates and updated ciphers. If a web page is accessible via HTTP it should be configured so, that all requests must be redirected to HTTPS. If a site has subdomains they also must be examined and ensured how properly they work under an HTTPS connection [19].

Record: "Strict-Transport-Security:" from HSTS directive instructs the web browser, that after that header, every connection to this page (including subdomains) must be granted using HTTPS. This directive will be active for 63072000 seconds after the browser gets it. If there is no way for an HTTPS connection, according to the directive, the connection will be dropped. Google has recommendations regarding "mag-age" parameters. They said that in a deployment process, "max-age" should be divided into 3 stages: 5 min; 1 week, and 1 month. Developers should monitor the metrics of the site, and fix any issues that come up once a developer is confident that there will be no problems, "max-age" should be increased to 2 years (63072000 seconds) [21].

HSTS protocol is additional protection in point of certificates. In a default case, when the CA is expired, and is not valid or the web browser gives a warning, the connection can be eavesdropped. But, in case of a correctly configured HSTS mechanism, the web browser will not let you access this website (unless you manually remove the page from the HSTS list).

There is one very important point that should be mentioned, if you provide an HSTS header for www.yourpage.com, it will cover only www.yourpage.com but not yourpage.com. This is a common mistake regarding the HSTS configuration. You should include a call for the base domain, in this case for yourpage.com, and add the

"includeSubDomains" parameter for proper protection [22].

If you are going to use Google HSTS preload service, requirements described by Google must be continuously satisfied by the webpage. If you remove the HSTS directive from the header, the web page will be automatically submitted for removal form (from the HSTS preload service). Google notifies users, that requesting or removing HSTS preload service may take a long time. It needs some time to reach new hardcoded updates to users.

In addition, browsers give the ability to manually add (or remove) web pages into the internal HSTS list. Open the browser and in the URL field type: "chrome://net-internals/#hsts". In the place of "chrome," you should type your browser vendor. But, generally, the browser will automatically correct the first parameter in this URL [26–29].

4. Conclusions

Web services have become significantly popular over the decade. Therefore, web-related threats have been increased. Security of the websites is very crucial for end users. So, before deployment of the security policies, it is very important to assess website security on the market. Thus, research was done on the point of HSTS protocol eligibility. According to the study, only 1% of Georgian websites are correctly configured on HSTS and are eligible for preload service (encrypted and authenticated). Results show, that in 36% of cases of authorization or transaction, sensitive data is transferred as a clear text and can be sniffed without significant effort of intruders. 37% of them are operated by state institutions or structures and 63% are private business operators.

In most cases, security policies are violated because of misconfiguration. Research shows, that in 77% of websites with a T/A module auto redirection from HTTP to HTTPS is not available. According to the study, only 21% of Georgian web pages have HSTS directives in the header. So, in 79% of using web pages, clients are at high risk of MITM attack. It is very interesting, that 73% of them have HTTPS support, but because of missing HSTS directives, they are vulnerable to SSL Strip attacks. Somehow, the security of subdomains is left behind by developers. Following this research, only 43% of Georgian websites serve all subdomains under HTTPS.

To conclude overall research, Georgian websites have critical security issues. Most of them are caused by misconfiguration. Submission for Google preload service takes too much time and has preliminary requirements. Still, there is another way for HSTS protection. Users can manually add the website to the internal HSTS list of browsers.

HSTS protocol does not guarantee full protection. There will be always another weakness, the "open door" for attackers, but we should add extra protections to minimize web-based vulnerabilities. For example, an NTP (Network Time Protocol) attack is used to compromise the HSTS mechanism. The next step will be checking how Georgian websites are protected against NTP attacks. Hence, cyber security experts should always work to strengthen the security of communication. All websites should satisfy the following requirements:

- Operate under a valid certificate.
- Operate only on HTTPS and should automatically redirect every HTTP request to HTTPS.
- All subdomains should be accessible only via HTTPS.
- A web page should be distributing properly configured HSTS directives.

As HSTS protects users and servers from data breaches (like user credentials, authorization parameters, personal information, and so on), the HSTS mechanism for webpages with a T/A module should be required by regulations of "Information Security" and "Personal Data Protection."

For a future work it is very important to do research and find solutions regarding different problems identified during this study, like certificates, redirections, authorization and so on. In case of Georgian webpage market, there is a tendency that should be noticed: when user enters wrong credentials, in some cases webpage returns specific error ("invalid username";

"incorrect password). It should not be specified, where mistake was made, in username or in the part of password. It critically increases risk of brute-forcing. So, for our future plan, other security elements of websites should be checked and find solution how to improve website security of Georgian market.

5. Acknowledgments

The work was conducted as a part of PHDF-19-519 financed by the Shota Rustaveli National Science Foundation of Georgia.

References

[1] V. Grechaninov, et al., Formation of Dependability and Cyber Protection Model in Information Systems of Situational Center, in: Workshop on Emerging Technology Trends on the Smart Industry and the Internet of Things, vol. 3149 (2022) 107–117.

[2] P. Anakhov, et al., Increasing the Functional Network Stability in the Depression Zone of the Hydroelectric Power Station Reservoir, in: Workshop on Emerging Technology Trends on the Smart Industry and the Internet of Things, vol. 3149 (2022) 169–176.

[3] V. Sokolov, P. Skladannyi, H. Hulak, Stability Verification of Self-Organized Wireless Networks with Block Encryption, in: 5th International Workshop on Computer Modeling and Intelligent Systems, vol. 3137 (2022) 227–237.

[4] V. Grechaninov, et al., Decentralized Access Demarcation System Construction in Situational Center Network, in: Workshop on Cybersecurity Providing in Information and Telecommunication Systems II, vol. 3188, no. 2 (2022) 197–206.

[5] V. Buriachok, et al., Invasion Detection Model using Two-Stage Criterion of Detection of Network Anomalies, in: Workshop on Cybersecurity Providing in Information and Telecommunication Systems, vol. 2746 (2020) 23–32.

[6] J. Hodges, C. Jackson, A. Barth, HTTP Strict Transport Security (HSTS). RFC, Proposed Standard (2012).

[7] Google. Online Checking Platform: Google HSTS Preload Service. https://hstspreload.org/.

[8] Z. Hu, et al., Statistical Techniques for Detecting Cyberattacks on Computer Networks based on an Analysis of Abnormal Traffic Behavior, International Journal of Computer Network and Information Security 12(6) (2020) 1–13.

[9] P. Razmjouei, et al., Ultra-Lightweight Mutual Authentication in the Vehicle Based on Smart Contract Blockchain: Case of MITM Attack, IEEE Sensors J. doi: 10.1109/JSEN.2020.3022536.

[10] Z. Hassan, et al., Detection of Distributed Denial of Service Attacks Using Snort Rules in Cloud Computing & Remote Control Systems, in: IEEE 5th International Conference on Methods and Systems of Navigation and Motion Control (2018) 283–288.

[11] M. Yaseen, et al., MARC: A Novel Framework for Detecting MITM Attacks in eHealthcare BLE Systems. J. Med. Syst. 43(324) (2019). doi: 10.1007/s10916-019-1440-0.

[12] M. Iavich, et al., The Novel System of Attacks Detection in 5G, Lecture Notes in Networks and Systems 226 (2021) 580–591.

[13] J. Kang, et al., Hybrid Routing for Man-in-the-Middle (MITM) Attack Detection in IoT Networks, in: 29th International Telecommunication Networks and Applications Conference (2019) 1–6. doi: 10.1109/ITNAC46935.2019.9077977.

[14] A. Prasad, S. Chandra, Defending ARP Spoofing-based MitM Attack using Machine Learning and Device Profiling, in: Int. Conf. on Computing, Communication, and Intelligent Systems (2022) 978–982, doi: 10.1109/ICCCIS56430.2022.10037723.

[15] B. G. Raúl, A. M. L. Sevillano, Services cloud under HSTS, Strengths and weakness before an attack of man in the middle MITM, in: Congreso Internacional de Innovacion y Tendencias en Ingenieria (2017) 1–5. doi: 10.1109/CONIITI.2017.8273322.

[16] Z. Hu, et al., High-Speed and Secure PRNG for Cryptographic Applications, Int. J. Comp. Netw. Inf. Secur. 12(3) (2020) 1–10.
[17] K. Krombholz, et al., If HTTPS Were Secure, I Wouldn't Need 2FA, End User and Administrator Mental Models of HTTPS, in: IEEE Symposium on Security and Privacy (2019) 246–263. doi: 10.1109/SP.2019.00060.
[18] S. Gnatyuk, et al., New Secure Block Cipher for Critical Applications: Design, Implementation, Speed and Security Analysis, Advances in Intelligent Systems and Computing 1126 (2020) 93–104.
[19] S. Wibowo, Enriching Digital Government Readiness Indicators of RKCI Assessment with Advance Https Assessment Method to Promote Cyber Security Awareness Among Smart Cities in Indonesia, in: Int. Conf. on ICT for Smart Society (2018) 1–4. doi: 10.1109/ICTSS.2018.8549974.
[20] S. Gnatyuk, et al., Method of Algorithm Building for Modular Reducing by Irreducible Polynomial, in: 16th Int. Conf. on Control, Automation and Systems (2016) 1476–1479.
[21] S. Gnatyuk, et al., Critical Aviation Information Systems: Identification and protection, Cases on Modern Computer Systems in Aviation (2019) 423–448.
[22] Z. Hu, et al., Method of Searching Birationally Equivalent Edwards Curves over Binary Fields, Advances in Intelligent Systems and Computing 754 (2019) 309–319.
[23] B. Li, et al., The Weakest Link of Certificate Transparency: Exploring the TLS/HTTPS Configurations of Third-Party Monitors, in: 18th IEEE Int. Conf. On Trust, Security and Privacy In Computing and Communications / 13th IEEE Int. Conf. on Big Data Science and Engineering (2019) 216–223. doi: 10.1109/TrustCom/BigDataSE.2019.00037.
[24] M. Iavich, et al., Hybrid Encryption Model of AES and ElGamal Cryptosystems for Flight Control Systems, in: IEEE 5th Int. Conf. on Methods and Systems of Navigation and Motion Control (2018) 229–233.
[25] B. Wang, et al., MTBD: HTTPS Tunnel Detection based on Multi-dimension Traffic Behaviors Decision, in: IEEE 24th Int Conf on High Performance Computing & Communications; 8th Int Conf on Data Science & Systems; 20th Int Conf on Smart City; 8th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (2022) 474–481. doi: 10.1109/HPCC-DSS-SmartCity-DependSys57074.2022.00091.
[26] O. Oksiiuk, V. Chaikovska, A. Fesenko, "Security Technique for Authentication Process in the Cloud Environment," in: IEEE Int. Sci.-Practical Conf. Problems of Infocommun. Sci. and Technol. (2019) 379–382. doi: 10.1109/PICST47496.2019.9061248
[27] L. Jiao, et al., CCSv6: A Detection Model for DNS-over-HTTPS Tunnel Using Attention Mechanism over IPv6, in: IEEE Symposium on Comput. and Commun. (2023) 1327–1330, doi: 10.1109/ISCC58397.2023.10218057.
[28] S. Špaček, et al., HTTPS Event-Flow Correlation: Improving Situational Awareness in Encrypted Web Traffic, in: NOMS 2022-2022 IEEE/IFIP Network Operations and Management Symposium (2022) 1–7. doi: 10.1109/NOMS54207.2022.9789877.
[29] R. Majumder, S. Datta, M. Roy, An Enhanced Cryptosystem Based on Modified Classical Ciphers, in: 8th International Conference on Advanced Computing and Communication Systems (2022) 692–696. doi: 10.1109/ICACCS54159.2022.9785033.