Journal of Computer Science

Original Research Paper

Survey of Websites and Web Application Security Threats Using Vulnerability Assessment

[1]Vincent Appiah, [2]Michael Asante, [3]Isaac Kofi Nti and [4]Owusu Nyarko-Boateng

[1,2]Deptartment of Computer Science, Kwame Nkrumah University of Science and Technology, Kumasi, Ghana
[3]Department of Electrical & Electronic Engineering, Sunyani Technical University, Sunyani, Ghana
[4]Department of Computer Science and Infomatics, University of Energy and Natural Resources, Sunyani, Ghana

Article history
Received: 14-03-2017
Revised: 17-01-2018
Accepted: 27-01-2018

Corresponding Author:
Vincent Appiah
West African Center for Cell Biology of Infectious Pathogens, University of Ghana, Ghana
Email: appiahv@rocketmail.com

Abstract: Nowadays information has become anasset to many institutions and as a result these institutions have become targets for people with malicious intents to attack these institutions. The web is now an important means of transacting business and without security, websites cannot thrive in today's complex computer ecosystem as there are new threats emerging as old ones are being tackled. Vulnerability assessment of websites is one of the means by which security can be improved on websites. This research seek to study and use vulnerability assessment as a tool to improve security by identifying vulnerabilities and proposing solutions to solve the security issues. Assessment was done on 5 web hosts belonging to different institutions in Ghana. Nmap, Nikto and Nessus were the tools used for the assessment, the assessment was carried out in four stages, and the first stage in the assessment was planning which involved activities and configurations performed before the actual assessment. The second stage was information gathering which involved obtaining information about the targets necessary to help identify vulnerabilities. This was followed by vulnerability scanning to identify vulnerabilities on the target hosts. The results indicated all the five hosts had security flaws which needed to be addressed. In all 16 vulnerabilities were identified on host 1, 8 vulnerabilities were identified on host 2, 15 vulnerabilities on host 3, 4 vulnerabilities on host 4 and 10 vulnerabilities on host 5. After the vulnerabilities were identified, a solution was proposed to mitigate the security flaws identified.

Keywords: Website-Security, Web-Application-Security, Network-Security, Protection-Tools, Firewall, Intrusion-Detection-System, Web-Security-Scanners, Web-Security-Vulnerability, Web-Vulnerabilities, Unauthorized-Access

Introduction

Website, web application and internet security is noteworthy area of research that affect a very wide range of computer users. Computer Security is the protection of computing systems and the data that they store or access. Currently, computer security is one of the most talked about issues in computing. This is due to its importance in almost every computer system (Hesham and Mohammad, 2012; Johari and Sharma, 2012). A critical fact in web applications and Internet security is that a computer and its associated system cannot be 100% reliable and confident (Appiah *et al*., 2017). Website or web application Vulnerability on the internet may compromise all the sensitive data and continuously give report on damage and cost (Durai and Priyadharsini, 2014; Appiah and Nyarko-Boateng, 2017). Website and web applications such as educational website, governments' website, healthcare applications and financial applications interact with its backend (database) several times upon a client request and there is a compromised in the security of such website and web application it results in loss of information, financial loss, law suits and identity theft (Chaudhari and Vaidya, 2014). According to Web Application Security Consortium the security of website is used to collect users data and web applications are of most important, a report from Web Application Security Consortium shows that 49% of web application has a high severity level vulnerabilities and 13% are

exposed to security vulnerabilities automatically. This unsecure website and web scripting, sql Injection, security misconfiguration, cookie theft, self-propagating worm's attacks and session hijacking (Chaudhari and Vaidya, 2014).

Figure 1 shows a graph of vulnerabilities within a web application from 2010 to 2012. From the graph in Fig. 1 it can be seen that this vulnerabilities in web applications is in a rise from year to year. Computer security is now employed in every field which deals with information processing and data storage. The use of debit, credit and ATM cards, and authentication mechanism and information access all encompasses computer security to safe guard the activities computer users and system (Nti *et al*., 2017).

In other to maintain a productive computing environment, computer security should be a priority. Cyber-crime is on the increase across the globe and as such organizations should also protect their systems against such attacks (Twum *et al*., 2016). One way of ensuring protection is to identify such security flaws before the attackers do by conducting security tests and implementing solutions to mitigate such security problems. In this paper, the state of the art in five randomly selected Ghanaian schools and companies' websites and web application security are examined, with a goal of identifying security teething troubles within the selected systems using vulnerability assessment as a tool. Grouping of identify vulnerabilities under broad groupings built upon security stuffs that website and web application have to preserve has been carried out and a discussion of the roots of these vulnerability have also been carried out.

Models of Computer Security

The introduction of easy to use and sophisticated Information Technology (IT) tools, has made attacks on a computer system or network a piece of cake for even trainee attackers. There for the principles that are used to safeguard security of resources on computer systems or network (Models of Computer Security) must be implemented on every computer system or network. They are usually referred to as the C.I.A. triangle as shown in Fig. 2.

Figure 2 shows the C.I.A triangle with its three parts namely:

- Confidentiality
- Integrit
- Availability

Confidentiality

This aspect of the models ensures that authorized system users are granted with access to places or information, thus confidentiality ensures that sensitive information do not fall in the hands of wrong people and it also guarantees the secrecy of information (Nemati, 2008). It is a security measure which protects against the disclosure of information to parties other than the intended recipient. For example in military, confidentiality ensures that military tactics, technology, weapons and other top secret information are not exposed to the enemy. Businesses use confidentiality to keep trade secrets safe and also to protect customer information.

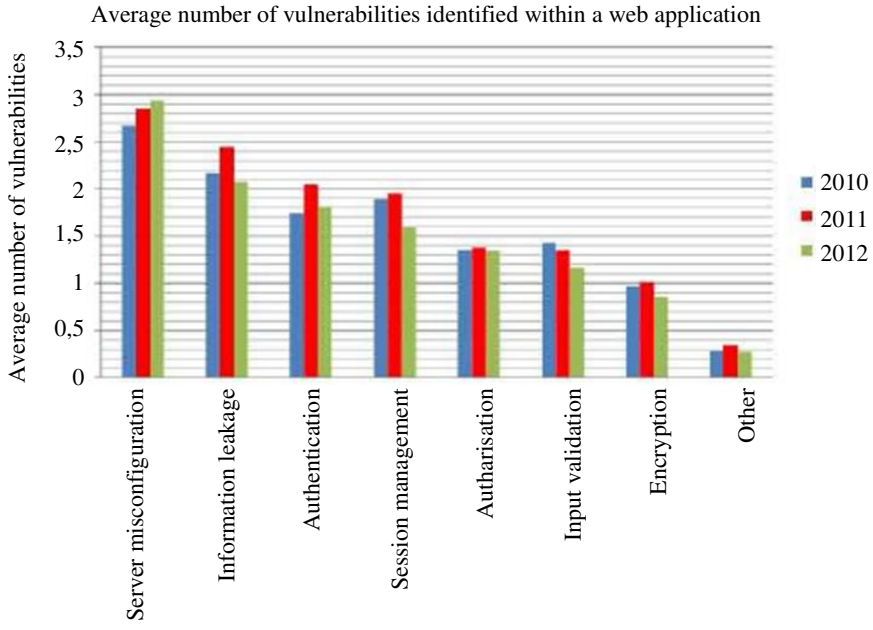

Fig. 1: Average number of vulnerabilities within web application (Source: Appiah *et al*., 2017)

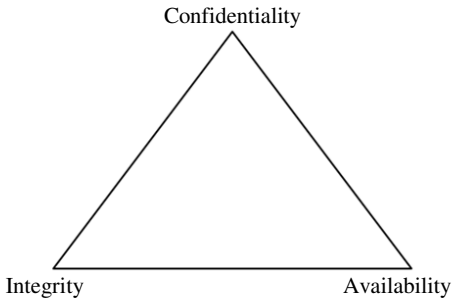

Fig. 2: Models of computer security

Confidentiality is breached if unauthorized individuals or applications can view information not intended for them (Whitman and Mattord, 2012). Some techniques employed to ensure confidentiality include the following:

- Encryption
- Biometric verification

Integrity

This ensures that data is accessed and modified by authorized users and as result makes data authentic and trustworthy. Integrity helps to prevent the changing of data in transit as this form of data is susceptible to modifications by unauthorized people. Data integrity is also a security measure that helps to ensure consistency in data by preventing its modification from unauthorized users and thereby making the data quality. Data with integrity is data which is unchanged and accurate from its source to its destination (Lehtinen Gangemi, 2011) attacks that can affect data integrity include man-in-the-middle attack where the attacker intercepts data in transit and makes changes to it before it reaches its destination. It must be noted that not all data modifications are intentional. Data modifications can be accidental.
Data modified can happen in the following ways:

- Natural disasters such as earthquakes, floods and fires
- Modification or corruption by computer viruses and worms
- Man-in-the-middle attacks
- Errors occurring during the transmission of data
- Faulty hardware such as storage disks

Techniques that can be used to ensure integrity include:

- Mirroring
- Access control

Access control is a security technique used to manage user access to information and resources on a computer system. This is usually done by means of identification where the user avails his/her credentials, authentication where the user confirms his/her identity and authorization where the user is granted access to the requested resources based on the permissions assigned to the user. With the aid of access control mechanisms, a user's access can be granted or revoked depending on the security policies for the system. File permissions and data privileges are examples of access control mechanisms used to ensure data integrity (Vacca, 2009).

Availability

This ensures that data is available to users at all times as well as preventing the loss of such data. Implementing availability also means that authorized users will always have access to their respective data even in emergency situations. Despite efforts being made to make data available some challenges are always encountered.

These challenges include:

- Occurrence of natural disasters such as floods and earthquakes
- Faulty equipment's
- Software errors
- Denial of service attacks

Regular backup can help ensure that data is always available.

Importance of Computer Security

Cyber security ensures that networks and computer systems are protected from cyber criminals. Having a secured network will prevent attackers from intruding and obtaining sensitive information as well as causing mayhem. An ever increasing cybercrimes requires that computer systems are well protected to prevent them from being attacked by cyber criminals (Hesham and Mohammad, 2012).

Computer security has now made it possible to safeguard information. Most transactions that are done today on web applications involve personal and sensitive information which if not protected might be exposed to third parties (Vandana *et al*., 2014).

Whenever there is a security breach, lots of money is spent to repair such breaches as well as improve them. Computer security ensures that such breaches are prevented thereby saving cost.

Computer security also helps to identify security flaws and vulnerabilities and appropriate solutions given. This is usually done through security audits, vulnerability assessments and penetration tests.

Website Security Risks

Websites now face a great deal of security risks. These risks can affect confidentiality, integrity or availability of data. Negative impact of some of these risks is very low while others can be very devastating. Some of the security risks are:

- Buffer overflows.
- Denial of service attacks (Dos)
- OWASP Top 10

Website and Web Applications

Figure 3 shows the basic business logic of a website and an internet application which has the client interface and the server end on a webserver and made known by a Uniform Resource Locator (URL). The internet server is understood by its name. The browser (client) and server talk via a transport protocol TCP. Figure 3 shows the fundamental architecture of data flow in website and a web application. The transport protocol is HTTP; the data format is Cascading Style Sheets (CSS) and hypertext mark-up language (HTML). The user click or enters a URL to call the application or access the website (Vandana *et al*., 2014). A request via communication protocol is sent to the server from the clients. A script at the net server removes input from the consumer knowledge and creates a request to a backend application server, e.g. amysql query to a database. The result is received from the backend by the webserver and returns a hypertext mark-up language (HTML) result page to the consumer. The result is displayed as a page by theclient"s browser. To show a page, the browser creates an interior picture for it.

Weakness of the Web Environment

Ten security risks has also been identified by Open Web Application Security Project (OWASP) as the most critical security risks associated with web applications. These risks are known to be common forms of attacks. Aside that they are known to be exploitable and can have a negative impact on websites when executed hence their rank as the top 10. The top 10 risks as published by OWASP are:

- Injection flaws
- Broken authentication and session management
- Cross site scripting
- Insecure direct object references
- Security misconfiguration
- Sensitive data exposure
- Missing level access control
- Cross Site Request Forgery (CSRF)
- Using components with known vulnerabilities
- Unvalidated redirects and forwards

Website and Web Application Vulnerability Issues

Research on web application vulnerability have shown that 50% of all web applications is not secure, whiles other works shows all are not secure and other work also established that 80% of the web applications has one critical security threats (Hesham and Mohammad, 2012). In another research work, shows that 49 percent of web application has a high severity level vulnerabilities and 13 percent are exposed to security vulnerabilities automatically (Durai and Priyadharsini, 2014; Chaudhari and Vaidya, 2014). In a report by (Acquaye, 2014) say that the government of Ghana official portal, which hosts fifty-Eight (58) websites of bureaus, departments and agencies was hacked by some unknown hacker and 11 website out of the 58 was under attached and substitute with a picture bearing a statement which reads “On us, the sword withdrawal of our homeland, unless entered, unless long suffering nation, unless anyone of us does damage to our homeland against our religion a bad idea to have all of the countries of virtual war will be opened in the Turks and tested my patience.” The report attributed the hacked a software failure and vulnerability on the part of some webmaster and administrators to bring up to date their software. The Reports further indicated that the attacked was the 2nd time in 3 years that hackers have taken over the government website. This disastrous occurrences in Ghana has raised many queries with regards to the security of country’s cyber space (Acquaye, 2014). A leading global cybersecurity solution (Trend Micro Incorporated) report in its 2016 annual reports titled “2016 Security Roundup: A Record Year for Enterprise Threats,” shows that the year 2016 had the highest online extortion issues as compared to the previous year (BiztechAfrica, 2017). The report said that Cyber threats in 2016 reached an all-time high, with corporate or Business Email Compromise (BEC) and ransomware scams attaining popularity increased among cybercriminals looking to extort corporate, business, or enterprises (BiztechAfrica, 2017). In this same report, a total of $1 billion losses resulted due to 748% increase in new cases of ransomware attacks.

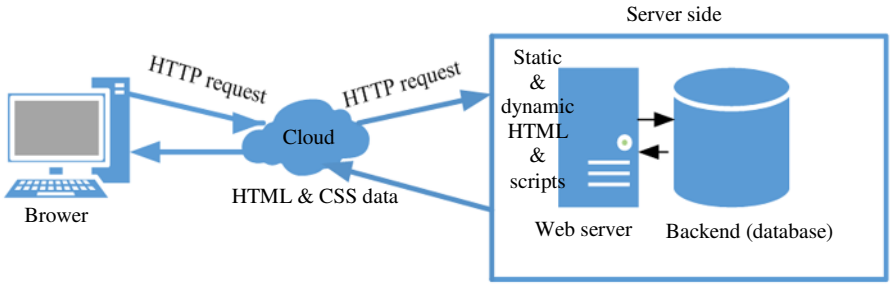

Fig. 3: Architecture of website/web application

The report also revealed that, a joint collaboration between Trend Micro and the Zero Day Initiative (ZDI) revealed seven hundred and eighty (780) vulnerabilities in 2016, and 678 out the total were carried to ZDI via their bug bounty program. An assessment made in the same report between vulnerabilities discovered in 2015 and 2016 shows that, Apple saw a 188% rise in vulnerabilities, whereas Microsoft viruses declined by 47% and also, the usage of fresh vulnerabilities in exploit kits fell by 71% (BiztechAfrica, 2017). "As threats have diversified and grown in sophistication, cybercriminals have moved on from primarily targeting individuals to focusing on where the money is: Enterprises," said Ed Cabrera, chief cybersecurity officer for Trend Micro (BiztechAfrica, 2017). In August 2013 a well-known mail service provider (yahoo) has it sheared of cybercrime attack. Yahoo experienced a major data breach in history, resulting in compromising One billion information of its account users' (BiztechAfrica, 2017). In another report, shows that the website of the Electoral Commission (EC) of Ghana was under attacked by unknown hackers with the intention to change the electoral results with "fake results" of the just ended election conducted by the commission in December 2016, but the commission said the attacked did not materials even though the site went down for some period (BBC, 2017).

The above discussion reveals that website and web application security is an endlessly moving target. New website and web application springs-up daily, fresh codes are released constantly, new web technologies are developed and adopted every single day; this causes new attacking techniques to be frequently released that can put every online business at a risk. In order to stay safe and protected, business, organizations, enterprises must receive timely information about how the most efficient way they can defend their websites against this attacks as the spring-up, gain visibility into the performance of their security programs, and also learn to compare with their industry peers. Tracking down these understandings is vital in order to stay ahead and truthfully improve enterprise website security. Hence this paper seeks to webmaster and administration of websites to identify security problems within the websites and web application systems, give suggestions for correction to improve their web-security. This research will also provide useful information on the selected websites for further research.

Importance of Vulnerability Assessment

Assessment of vulnerability in a website and web application will unleash the following importance:

- It helps in the identification of vulnerabilities and threats
- Information obtained can be used to improve security of computer systems
- It also helps to know the architecture of the computer system
- Security breach can be expensive. Vulnerability assessment helps prevent such breaches through identification of security flaws

Tool and Methods

Tools

The following tool and software were employed for this research:

- Oracle Virtual Box
- Nmap
- Nikto
- Nessus

Methods

The Vulnerability assessment was conducted on 5 websites belonging to different organisations in Ghana. The IP's, URL address of this organizations are exempted from the paper for security reasons. Instead aliases were used. The webservers were categories from one (1) to five (5) as their names. Four phase assessment was carried, namely:

- Planning
- Information Gathering
- Vulnerability scanning
- Analysis and Reporting

Figure 4 shows the flow chart of the assessment methodology employed by this research.

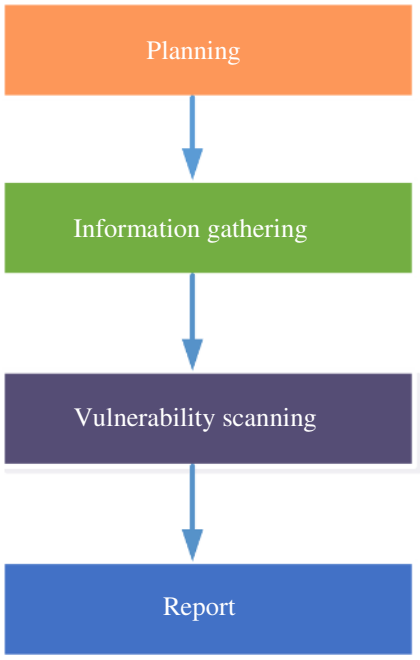

Fig. 4: Vulnerability assessment methodology

Planning

The planning phase consisted of all activities that we needed to be performed before the actual vulnerability assessment is performed. In the planning phase, the scope and objectives for the activity is defined. It also involves getting Management Approvals and signing of documents. Also the testing team prepares a strategy for the performance of the assessment based on security policies of the requesting organization, industry standards and best practices.

Information Gathering

At this stage the team centered on gathering as much information as possible about the target websites which will be helpful in the finding of vulnerabilities. The Graphic User Interface (GUI) version of Nmap known as Zenmap was used to perform information gathering tasks, which gave us the IP address identification, port scanning and web application fingerprinting.

Vulnerability Scanning

At this stage the vulnerabilities and weakness in the selected websites were identify. Results from the information gathering was used to initiate the vulnerability scan of the web hosts. Nessus and Nikto were used for the vulnerability scanning on all the 5 web hosts. Testing was done to identify vulnerabilities as suggested by the OWASP Guidelines. The reported generated included the following:

- Name of vulnerability: This helps when the tester wants to search for additional information of the vulnerability
- Vulnerability description: A brief description of the vulnerability is also included as well as how it can be exploited by attackers
- The risk factor of vulnerability: This tells the severity of the identified vulnerabilities and helps in prioritization of solution for the vulnerabilities identified
- Reference: This indicates the vulnerability ID and related vulnerabilities as well as which database it can be found.
- Plugin: This indicates which plugin was used to identify the vulnerability

Results and Discussion

The results of vulnerability assessments performed on selected web hosts and the significance of the findings is presented.

Results

In a total host 1 had the highest vulnerabilities of 16, followed by host 3 with 15 vulnerabilities, host 5 with 10 vulnerabilities and host 2 with 8 vulnerabilities and host 4 with 4 vulnerabilities. A graphical representation is as shown in Fig. 5.

Vulnerabilities Analysis of Host 1

Out of the 16 vulnerabilities identified, 37.5% was classified as medium risk, while 25% were low. 6 vulnerabilities were also labelled as info. No vulnerability was classified as critical or high. The threat level for the server could be said to be medium since the most severe among the vulnerabilities were labelled medium as shown in Fig. 6.

Vulnerabilities Analysis of Host 2

Figure 7 shows the vulnerabilities analysis of host 2. Out of the 8 vulnerabilities identified in host 2, the risk posed by 3 of them was medium while the number of vulnerabilities with risk factors low and info were 2 and 3 respectively. There was no vulnerability associated with risk factors critical and high. Threat level for this host was also medium because the most severe among the identified vulnerabilities was medium.

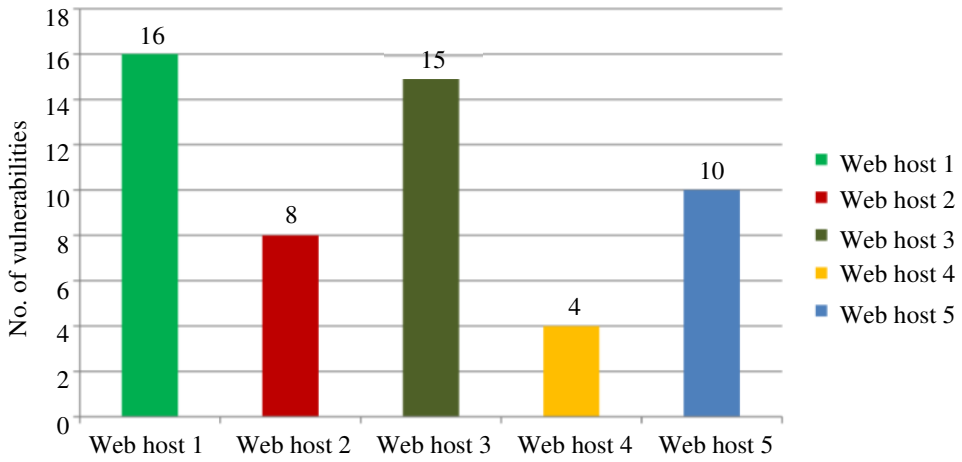

Fig. 5: Vulnerability summary for the scanned web hosts

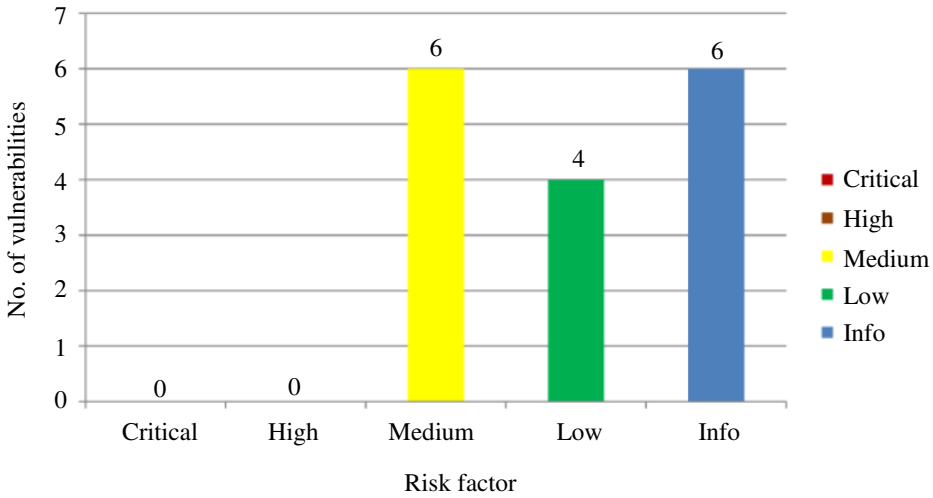

Fig. 6: Vulnerabilities Analysis on host 1

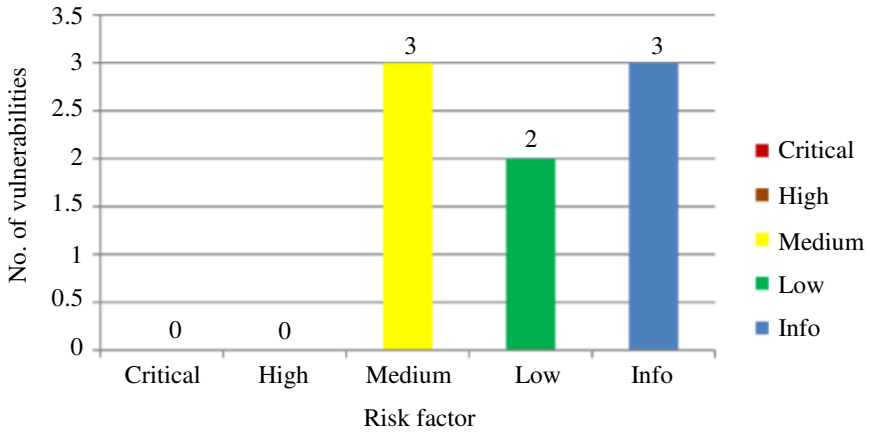

Fig. 7: Identified vulnerabilities on host 2

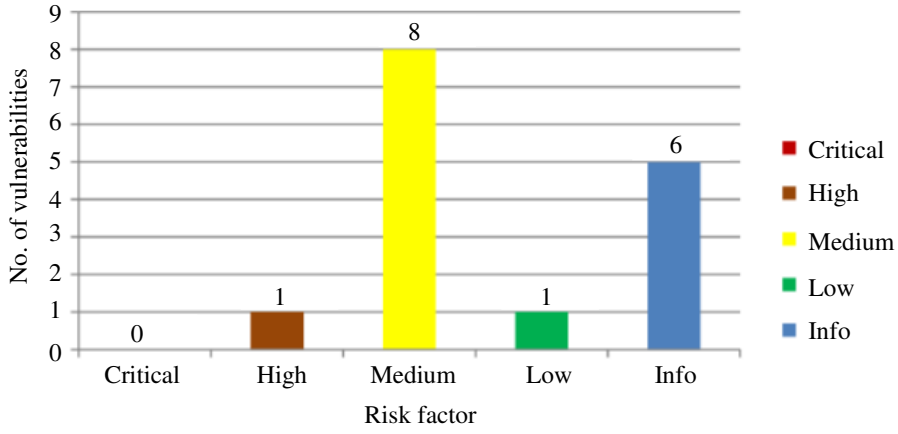

Fig. 8: Vulnerabilities on host 3

Vulnerabilities Analysis of Host 3

Figure 8 shows the vulnerabilities analysis of host 3, out of the 15 vulnerabilities identified in host 2, the severity of 1 of them was high. The risk posed by 8 of them was medium while the number of vulnerabilities with risk factors low and info were 1 and 5 respectively. No vulnerability was identified as critical. The most severe was labelled High making its thread level as High. Therefore in dealing with the identified vulnerabilities, the vulnerability labelled high should be tackled first.

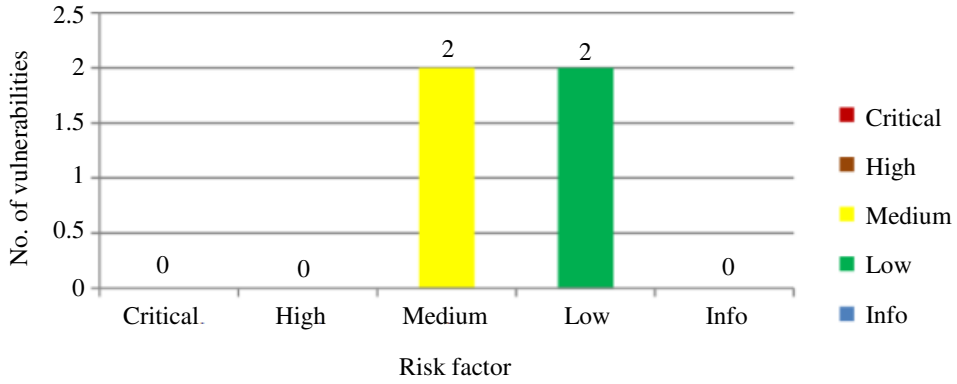

Fig. 9: Vulnerabilities on host 4

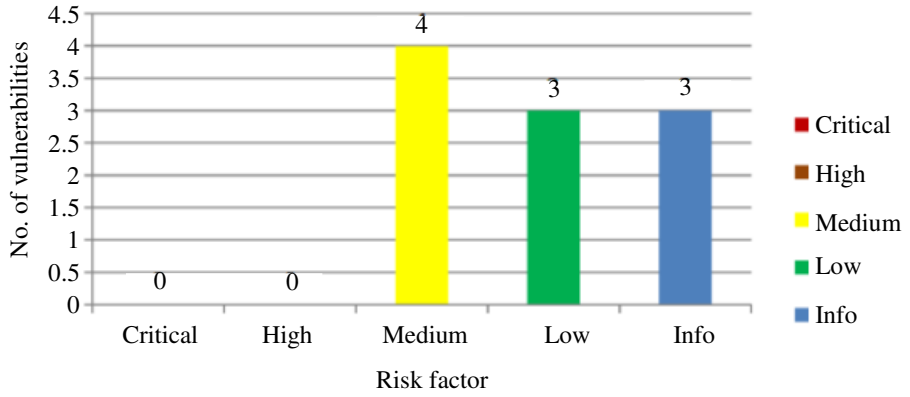

Fig. 10: Vulnerabilities on host 5

Table 1: Identified vulnerabilities

Vulnerability	Host 1	Host 2	Host 3	Host 4	Host 5
Anonymous FTP enabled			√		
Apache HTTP server user dir directive					
Username enumeration			√		
AutoCompletion of password	√				
Cleartext transmission of sensitive Information	√			√	
Clickjacking vulnerability	√	√	√	√	√
Cookies without HTTPonly flag identified	√	√	√		√
Cross-site scripting vulnerability			√		
Directory indexing enabled					√
DOS Amplification Vulnerability			√		
FTP bounce attack vulnerability			√		
HTTP TRACE method enabled.		√			√
Logjam vulnerability.	√				

Vulnerabilities Analysis of Host 4

Figure 9 depicts the vulnerabilities analysis of host 4. Host 4 recorded the least number of vulnerabilities. Out of 4 vulnerabilities were identified, 2 of the vulnerabilities had a severity of medium while 2 vulnerabilities were labelled as low. No vulnerability was classified as critical, high or info. Threat level was therefore medium.

Vulnerabilities Analysis of Host 5

Figure 10 shows the vulnerabilities analysis of host 2, 10 vulnerabilities were identified in host 5. 4 of them had a risk factor of medium. 3 vulnerabilities had risk factor low and 3 vulnerabilities were labelled as informational (info). The most severe vulnerabilities were labelled medium and so the threat level was also medium.

Table 2: Identified vulnerabilities

Microsoft Internet Information Services (IIS) Flaw				√	
Missing HSTS	√	√	√		
MoinMoin Two Unspecified XSS.		√	√		
Multiple	Web	Server	Default	Page	
Fingerprinting weakness	√		√		
Multiple web server interesting web					
Document	√		√		√
Multiple web server robots.txt remote					
Information disclosure	√				
PHP expose_php information disclosure					√
RC4 algorithm invariance-weakness	√	√	√		
RSA keys less than 2048 bits	√			√	
SMTP service supports cleartext login			√		√
SSH Protocol CBC mode enabled	√	√			√
SSH weak mac algorithm enabled	√	√			√
SSL v2 and SSL v3 detection	√		√		
Untrusted SSL-Certificate	√				
Usr/doc directory information disclosure			√		√
Weak hashing algorithms for the Signing of SSL certificate	√				

Identified Vulnerabilities

In all, 28 vulnerabilities were discovered in the 5 hosts that were scanned. The identified vulnerabilities and the associated hosts are listed in Tables 1 and 2.

Discussion

This study was done to identify vulnerabilities in selected web hosts and a solution proposed to mitigate the identified vulnerabilities. Even though everything was done to ensure that the tests and methodology used followed standard procedures, certain factors affected the results of the tests.

One of the factors was the fact that the websites were productive environments and as such there was a high risk of disrupting the services. Due to this, Denial of service (Dos) tests as well as memory corruption tests were not performed. This affected the results because even though security issues were discovered, there was no way of investigating whether the websites were secure from Dos attacks and memory corruption attacks.

The vulnerability assessment was performed outside the network. This was to simulate how an attacker outside the network might infiltrate the network using discovered vulnerabilities. It is therefore possible that if internal tests were done, more information would have been discovered since attacks can come from within the network. However no internal assessment was done. This factor should therefore be considered when doing an analysis on the result.

This assessment was done on 5 websites belonging to different entities. At the end of the assessment security issues were discovered. The owners of the websites were not aware of these issues and so this assessment was useful to them.

Vulnerabilities were discovered in all the 5 scanned web hosts. Some of the vulnerabilities existed in more than one of the scanned hosts while others were identified on a single host. From the report generated by Nessus and Nikto, the identified vulnerabilities were due to the following reasons:

- Cryptographic flaws
- Security misconfigurations
- Applications with vulnerabilities

Cryptographic Flaws

Figure 11 shows a typical cryptographic flaw. Attacks such as logjam and FREAK were successful due to existing cryptographic flaws in the target web applications.

The number of cryptographic flaws identified on host 1 was 6 whiles host 2 and 3 had 3 and 2 flaws respectively. 1 cryptographic flaw was identified on host 4 and 3 flaws identified on host 5.

The following cryptographic flaws were identified:

- Logjam vulnerability
- RC4 Algorithm Invariance-Weakness.
- RSA keys less than 2048 bits
- SSH Protocol CBC Mode Enabled.
- SSH weak Mac Algorithm enabled
- Weak Hashing Algorithms for the Signing of SSL Certificate

Security Misconfigurations

The infrastructure that supports a Web application comprises a complex variety of devices and software, including servers, firewalls, databases and OS and

application software. All these elements need to be securely configured and maintained. If security configurations are not properly done the system can be compromised, which can lead to unauthorised access, modification and transmission of data, Denial of service, Virus attacks, Memory corruption attacks, Buffer overflow attacks and Installation of backdoors. It was discovered that, security misconfigurations existed on all the five host servers. Hosts 1, 2, 3 and 5 all had cookies set without httponly flag making them susceptible to man-in-the middle attacks. Host 2 and 5 had enabled HTTP TRACE method making them vulnerable to Cross-Site Scripting attacks. Host 5 had enabled directory indexing which could result in unauthorized access to information and data. Host 3 was also discovered to be running an ftp server which was not properly configured making it possible to login anonymously and also vulnerable DOS attacks. Host 3 remote DNS server had not been properly configured so could answer any request making vulnerable to DOS attacks. A php misconfiguration was also found on host 5 web server and this could allow information disclosure via the use of PHP Easter eggs. Also HTTP Strict Transport Security was found missing on hosts 1,2 and 3 making them vulnerable to man-in-the middle attacks since the http transmission was insecure. All the five hosts were also vulnerable to click jacking due to the absence of X-Frame Options Response Header and so attackers could render the content of the web pages they generate to steal sensitive information from users.

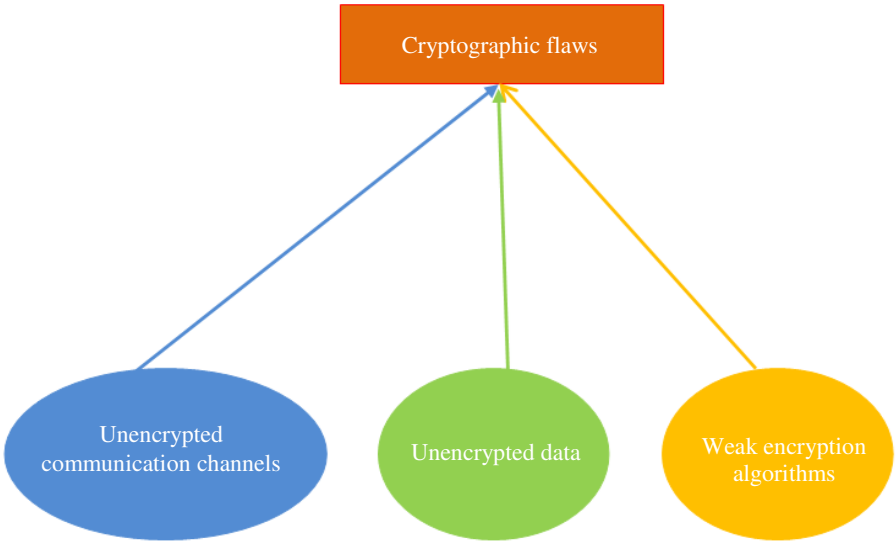

Fig. 11: Cryptographic flaws

Fig. 12: Web defacement of host 1

Fig. 13: Web defacement of host 2

Applications with Vulnerabilities

Insecure applications occur due to poor application design based on the false assumption that users will always follow the application rules. For example, if a user's account ID is shown in the page URL or in a hidden field, a malicious user may be able to guess another user's ID and resubmit the request to access their data, particularly if the ID is a predictable value. Common places where this data is incorrectly exposed are URLs and links, hidden form fields, the unprotected view state in ASP.NET, drop-down list boxes, JavaScript code and client-side objects like Java applets. Servers running applications with vulnerabilities can easily be compromised. The applications are also susceptible to attacks such as denial of service, memory corruption, buffer overflow and XSS attacks.

In this research, 3 of the web hosts were found to be running applications with known vulnerabilities. Host 4 was found to be running Microsoft IIS 7.0 which contained vulnerabilities such as Memory Corruption, Buffer Overflow and Denial of Service. Host 2 and 3 both were running MoinMoin 2 which also contains 2 vulnerabilities relating to Cross-Site Scripting.

Other Security Issues

Aside these issues, a search from the database of http://zone-h.org revealed that web host 1 and host 2 had been recently been compromised as shown in Fig. 12 and 13. This represented a serious information disclosure because the IP address as well as host operating system is publicly available. Owners of web host 1 and host 2 are therefore advised to review their security policies and fix the identified flaws to prevent another attack.

Conclusion and Recommendation

The vulnerability assessment was helpful as it provided information about the security of the selected websites. Vulnerabilities were discovered in all the web hosts that were scanned. Some of vulnerabilities were found in all the web hosts whiles others were specific to a particular host. These discoveries brought to light that there are security issues that need to be addressed in all the five hosts that were scanned. As technology is evolving, new techniques are also being developed to exploit computer systems. It is therefore important to be abreast with such techniques in other to combat these security threats.

In conclusion we notice that managerial issues or administration errors, such as the following contribute immensely to security threats:

- Problems associated with security are generally solved through short-term recovery, hence leading to the problems appearing again very quickly
- Mangers and webmaster of the institutions do not recognise that numerous security threats causes reduction of organization's reputation
- Mangers of website don't consider the fact that the data on their websites is cost money, in addition to losing the ability of estimate the information cost
- Most Mangers and webmaster depends on off the shell protection tool and software such as intrusion discovery system or firewall without doing regular monitoring them regularly and their websites

- Because most institutions want to launch or re-launch their website as quickly as possible, well trained technical men are not given the website development contract due to cost (cheap labour) forgotten that there is a saying that says "if you think education is expensive try ignorance "and enough time are not given for bugs fixing

Based on the findings, it is recommended that:

- All security issues identified should be resolved
- All applications being used on the respective web servers should be upgraded or changed to a more secure one
- All hosts within the respective networks should be checked for security flaws
- Regular tests should be conducted to access the security of the respective networks
- Personnel should be trained on how to maintain security of respective networks
- There should be internal vulnerability assessments for the websites
- Denial of service and Memory corruption tests must be done on the web servers

Future Work

The vulnerabilities identified in the five selected websites indicates the most websites are exposed to this vulnerabilities. In feature research, we seek to outline preventive maintenance scheduled and well-known techniques to secure websites owner and also educate most webmasters.

Acknowledgement

We express our thanks to God Almighty for His guidance and protections throughout the period of this research work and to all friends and colleagues who contributed to make this work a success we say God bless you.

Author's Contributions

Vincent Appiah: Data collecction and penetration test conception, design.

Michael Asante: Data analysis and editorial assistance.

Isaac Kofi Nti: Report writing and data analysis, participates in crafting.

Owusu Nyarko-Boateng: Literature review and reviewing.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and there are no ethical issues involved.

References

Acquaye, N.A., 2014. Software vulnerability led to Ghana government site hack.

Appiah, V., I. K. Nti and O. Nyarko-Boateng, 2017. Investigating websites and web application vulnerabilities: Webmaster's perspective. Int. J. Applied Inform. Systems, 12: 10-15. DOI: 10.5120/ijais2017451673

BBC, 2017. Ghana election commission website hit by cyber attack.

BiztechAfrica, 2017. Annual security roundup report, "2016 Security Roundup.

Chaudhari, X. and M. Vaidya, 2014. A survey on security and vulnerabilities of web application. Int. J. Comput. Sci. Inform. Technologies, 5: 1856-1860.

Durai, K. and K. Priyadharsini, 2014. A survey on security properties and web application scanner. Int. J. Comput. Sci. Mobile Comput., 3: 517-527.

Hesham, A. and S. Mohammad, 2012. Survey of web application and internet security threats. Int. J. Comput. Science Security, 12: 67-76.

Johari, R. and P. Sharma, 2012. A Survey on Web Application Vulnerabilities (SQLIA, XSS) Exploitation and Security Engine for SQL Injection. Proceedings of the International Conference on Communication Systems and Network Technologies, May, 11-13, IEEE Xplore Press, Rajkot, pp: 453-458. DOI: 10.1109/CSNT.2012.104

Lehtinen, R. and G.T. Gangemi, 2011. Computer Security Basics, 2nd Edn," O'Reilly.

Nemati, H., 2008. Information security and ethics: Concepts, methodologies, tools, and applications: Concepts, methodologies, tools and applications. IGI Global, pp: 73-75.

Nti, I.K., J.A. Ansere and A. Appiah, 2017. Investigating ATM frauds in sunyani municipality: Customer's perspective. Int. J. Sci. Eng. Appli., 6: 59-65. DOI: 10.7753/IJSEA0602.1006

Twum, F., K. Nti and M. Asante, 2016. Improving security levels in Automatic Teller Machines (ATM) using multifactor authentication. Int. J. Sci. Eng. Appli., 5: 126-134. DOI: 10.7753/IJSEA0503.1003

Vacca, J., 2009. Computer and information security handbook. Elsevier Inc., pp: 63-70.

Vandana, D., Y. Himanshu and A. Jain, 2014. Web application vulnerabilities: A survey. Int. J. Comput. Appli., 108: 25-31.

Whitman, M.E. and H. Mattord, 2012. Principles of Information Security, 4th Edn.

APPENDIX

Table 3: Nmap information gathering output

Web host	IP address
1	***.***.53.162
2	***.***.80.197
3	***.***.58.115
4	***.***.27.33
5	***.***.181.70

Table 4: IP addresses of the scanned websites

Web host	No. of open ports	No. of filtered ports	Closed
1	3	995	2
2	3	997	0
3	14	18	968
4	6	994	0
5	14	1	985

Table 5: Port scanning summaries

Port	Protocol	State	Service	Version	OS
53	tcptcptcptcptcp	Open closed open closed	Domain	ISC BIND Not Disclosed	Linux 3.2-3.6
80			http ident http svn	Apache httpd	
113					
443				Apache httpd	
3690					

Table 6: Information gathering details for Web host 1

Port	Protocol	State	Service	Version	OS
22	tcptcp	open	ssh http	OpenSSH 5.3(protocol 2.0)	Linux 3.1.9
80		open		Apache httpd 2.4.16 ((Unix) OpenSSL/1.0.1e-fips mod_bwlimited/1.4)	
				Apache httpd 2.4.16 ((Unix) OpenSSL/1.0.1e-fips mod_bwlimited/1.4)	
	tcp		http		
443		open			

Table 7: Information gathering summary for Web host 2

Port	Protocol	State	Service	Version	OS
1	tcptcptcptcptcptcptcpt	filtered filteredfiltered	tcpmuxcompressnet		
	cptcptcptcptcptcp	filteredfilteredfiltered	unknown unknown echo		
		filteredfilteredfiltered open	discard daytime qotdchargen		
3		filtered open open	ftp sshsmtpsmtp		
4					
6					
7					
9					
13					
17					
19					
21					Pure-FTPd
22					
25					Exim smtpd
26					
53	tcptcptcptcptcptcptcptcp	open	domain		
80		openopenopenope	http pop3	Apache httpd	
110	tcptcptcptcptcptcptcptc	nope	imap		
143		filtered	http smtpssmtp	Dovecot pop3d	
443			http-rpcepmapimap		
465		open open	rpcepmapimap	Dovecot imapd	
587		filtered filtered open	pop3		
593		filtered filteredfiltered	pvuniwienEtherN	Apache httpd	
		Open	et/IP-1 mysqldec-		
993		filtered filtered	notes unknown	Exim smtpd 4.85	

Table 7: Continue

995	unknown	
1081	http	
2222	unknown unknown	
3306		
3333		Dovecot imapd
5915		
6692		Dovecot pop3d
8080		
9575		MySQL 5.5.42-37.1
49165		Apache httpd

Table 8: Information gathering summary for Web host 3

Port	Protocol	State	Service	Version	OS
21	tcptcp	open open	ftp	Microsoft ESMTP	Microsoft
25	tcptcptcptcp	openopenopen	smtp	7.0.6002.18264	Windows
80			http		
443			https		
1032			iad3		
1248			hermes		

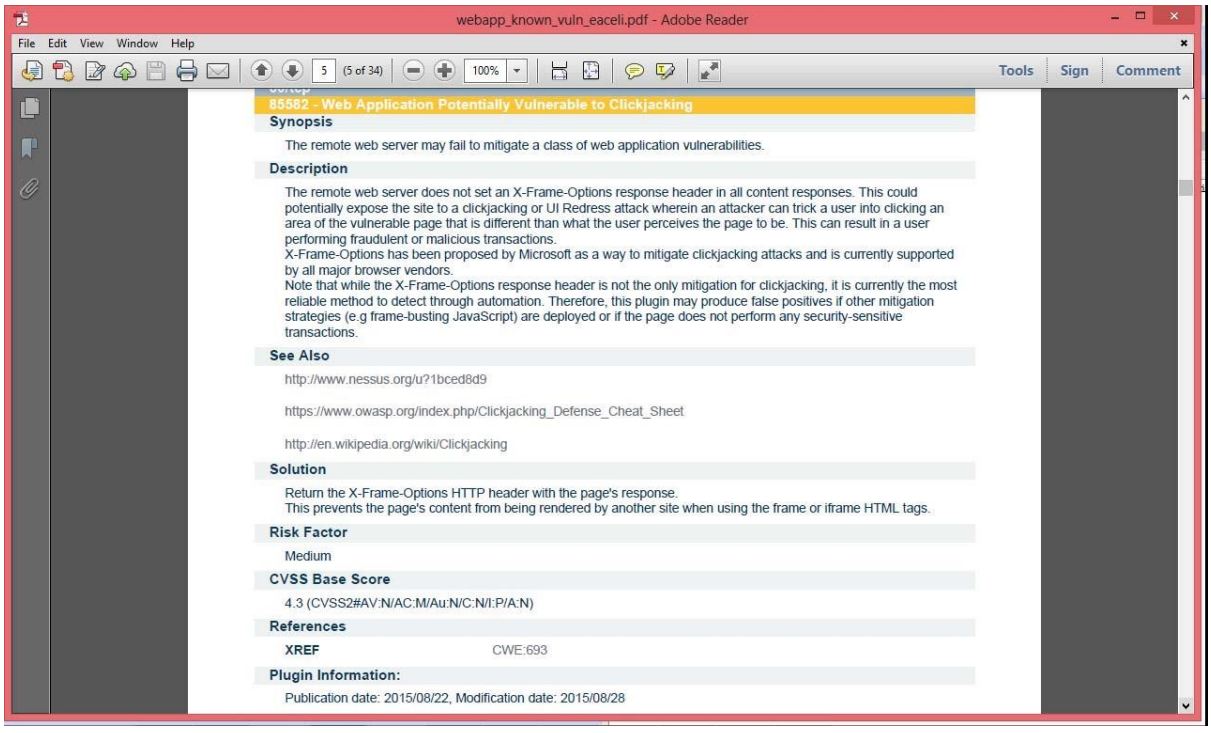

Fig. 14: Shows a sample of the report generated by Nessus