Research Journal of Engineering Sciences ______________________________ ISSN 2278 – 9472
Vol. **1(4),** 53-56, October **(2012)** *Res. J. Engineering Sci.*

Technical Problems Especially Web Security Related With World Wide Web

Kumar Santosh
Sainath University, Ranchi, INDIA

Available online at: www.isca.in
Received 4th September 2012, revised 1st October 2012, accepted 20th October 2012

Abstract

The World Wide Web (or the proper World-Wide Web; abbreviated as WWW or W3, and commonly known as the Web is a system of interlinked hypertext documents accessed via the Internet. With a web browser, one can view web pages that may contain text, images, videos, and other multimedia and navigate between them via hyperlinks. Many formal standards and other technical specifications and software define the operation of different aspects of the World Wide Web, the Internet, and computer information exchange. Many of the documents are the work of the World Wide Web Consortium (W3C), headed by Berners-Lee, but some are produced by the Internet Engineering Task Force (IETF) and other organizations. Usually, when web standards are discussed, the following publications are seen as foundational: Recommendations for markup languages, especially HTML and XHTML, from the W3C. These define the structure and interpretation of hypertext documents. Recommendations for stylesheets, especially CSS, from the W3C. Standards for ECMAScript (usually in the form of JavaScript), from Ecma International. Recommendations for the Document Object Model, from W3C. Security threats to web sites and web applications (webapps) come in many forms. Data centres and other assets used for hosting web sites and their associated systems need to be protected from all types of threat. Threats should be identified using application threat modelling and then evaluated with a vulnerability assessment. Vulnerabilities can be removed or reduced and countermeasures put in place to mitigate the effects of an incident should the threat be realized. Some of them are security policies, using technology, content filtering.

Keywords: Web Security, internet engineering task force.

Introduction

The World Wide Web (or the proper World-Wide Web; abbreviated as WWW or W3, and commonly known as the Web) is a system of interlinked hypertext documents accessed via the Internet. With a web browser, one can view web pages that may contain text, images, videos, and other multimedia and navigate between them via hyperlinks. Many formal standards and other technical specifications and software define the operation of different aspects of the World Wide Web, the Internet, and computer information exchange. Many of the documents are the work of the World Wide Web Consortium (W3C)[1] headed by Berners-Lee, but some are produced by the Internet Engineering Task Force (IETF) and other organizations.

Usually, when web standards are discussed, the following publications are seen as foundational: i. Recommendations for markup languages, especially HTML and XHTML, from the W3C. These define the structure and interpretation of hypertext documents. ii. Recommendations for stylesheets, especially CSS, from the W3C. iii. Standards for ECMA Script (usually in the form of Java Script), from Ecma International. iv. Recommendations for the Document Object Model, from W3C.

Additional publications provide definitions of other essential technologies for the World Wide Web, including, but not limited to, the following: i. Uniform Resource Identifier (URI), which is a universal system for referencing resources on the Internet, such as hypertext documents and images. URIs, often called URLs, are defined by the IETF's RFC 3986 / STD 66: Uniform Resource Identifier (URI): Generic Syntax, as well as its predecessors and numerous URI scheme-defining RFCs; ii. Hyper Text Transfer Protocol (HTTP), especially as defined by RFC 2616: HTTP/1.1 and RFC 2617: HTTP Authentication, which specify how the browser and server authenticate each other.

As email and web technologies converge, the number of security threats has grown, both in terms of creativity and effectiveness[2]. Web-based threats are proving to be a nightmare for IT administrators and computer users. Although technology helps to counter these threats, a more holistic approach is needed, one that includes strict and enforceable policies as well as a proper awareness program.

WS-Security (**Web Services Security, short WSS**) is a flexible and feature-rich extension to SOAP to apply security to web services. It is a member of the WS-* family of web service specifications and was published by OASIS. The protocol specifies how integrity and confidentiality can be enforced on messages and allows the communication of various security token formats, such as SAML, Kerberos, and X.509. Its main focus is the use of XML Signature and XML Encryption to provide end-to-end security[3].

WS-Security describes three main mechanisms: i. How to sign SOAP messages to assure integrity. Signed messages also provide non-repudiation. ii. How to encrypt SOAP messages to assure confidentiality. iii. How to attach security tokens to ascertain the sender's identity. iv. The specification allows a variety of signature formats, encryptions algorithms and multiple trust domains, and is open to various security token models, such as: X.509 certificates, Kerberos tickets, UserID/Password credentials, SAML-Assertion, Custom defined token. The token formats and semantics are defined in the associated profile documents.

WS-Security incorporates security features in the header of a SOAP message, working in the application layer. These mechanisms by themselves do not provide a complete security solution for Web services. Instead, this specification is a building block that can be used in conjunction with other Web service extensions and higher-level application-specific protocols to accommodate a wide variety of security models and security technologies. In general, WSS by itself does not provide any guarantee of security. When implementing and using the framework and syntax, it is up to the implementer to ensure that the result is not vulnerable. The generally used cases are Transport Layer Security (without WS-security), [edit] End-to-End security, Non-Repudiation, Alternative transport bindings, Reverse proxy/common security token etc [4].

Still the following may effect a great deal: i. If there are frequent message exchanges between service provider and consumer, the overhead of XML SIG and XML ENC are significant. If end-to-end security is required, a protocol like WS-Secure Conversation may reduce the overhead. If sufficient, use only encryption or signing, as the combination of both is significantly slower than the mere sum of the single operations[5]. See Performance below. ii. The merging of several XML-schemata like SOAP, SAML, XML ENC, XML SIG might cause dependencies on different versions of library functions like canonicalization and parsing, that are difficult to manage in an application server.

Security threats to web sites and web applications (web apps) come in many forms. Data centres and other assets used for hosting web sites and their associated systems need to be protected from all types of threat. Threats should be identified using application threat modelling and then evaluated with a vulnerability assessment[6]. Vulnerabilities can be removed or reduced and countermeasures put in place to mitigate the effects of an incident should the threat be realised. The main types of threats to web systems are listed below:

Physical threats include loss or damage to equipment through fire, smoke, water and other fire suppressants, dust, theft and physical impact. Physical impact may be due to collision or the result of malicious or accidental damage by people[7]. Power loss will affect the ability for servers and network equipment to operate depending upon the type of back-up power available and how robust it is.

Errors caused by people include operator/user error such as accidental deletion of data or destruction of software programs, configurations or hardware. The other major error caused by people is leaving weaknesses (vulnerabilities) in software. This can include escalation of privileges, authentication which can be bypassed, incorrect implementation of encryption, failure to validate input and output data, weak session management, failure to handle errors correctly, etc. Good programming practices can reduce the vulnerabilities which human error can exploit.

Both equipment and software malfunction threats can impact upon the operations of a website or web application. All assets required for the operation of the web system must be identified to be able to evaluate the threats. Malfunction of software is usually due to poor development practices where security has not been built into the software development life cycle.

Malware, or malicious software, comes in many guises. Web servers are popular targets to aid distribution of such code and sites which have vulnerabilities that allow this are popular targets.

Spoofing where a computer assumes the identity of another and masquerading where a user pretends to be another, usually with higher privileges, can be used to attack web systems to poison data, deny service or damage systems.

Scanning of web systems are usually part of network or application fingerprinting prior to an attack, but also include brute force and dictionary attacks on username, passwords and encryption keys. Monitoring of data (on the network, or on user's screens) may be used to uncover passwords or other sensitive data.

Examining 'found' data from accessible sources such as the network, search engines and waste. The actual target information could be found, but more often scavenging is used as a way to select other threats for vulnerabilities that are known to exist for the web system (e.g. operating system, firewall type, server software, application software). Overloading a system through excessive traffic can lead to denial of service for other users or system failure.

Network attack techniques such as tunnelling to access low level system functions can mean the target such as a router or server can be taken over. Once an attacker has control, this can be used to attack other assets required for the continued operation of a web site.

Phishing: The term refers to attacks where the victim is led to believe that he or she is on a legitimate website, when in fact it is just a copy of the real one. This attack relies on the fact that anyone can create their own website and any website can look like any other. Phishing attacks have been known to target company email websites (webmail), public email websites (like Gmail) and popular sites like Amazon or eBay. Users can identify a phishing website in a number of ways. The first is to look at the URL. Another effective prevention technique is to never follow links by email but to type them in or use bookmarks. Although not foolproof, these methods make it harder for attackers to pull off the scam.

Web browser exploits: Cybercriminals have also set up websites that exploit security holes in the web browser. This technique allows them to gain access without the victim's knowledge. Web browsers are complex software. They have to handle various file formats, such as images, sound and HTML, Javascript and a multitude of other technologies. All of these features add to the attack surface of the web browser, thus making the technology relatively weak from a security standpoint.

Third party add-ons: The majority of websites require the use of third party add-ons such as Adobe Flash player and Acrobat Reader. Both of these widely used products have become a favorite target for cybercriminals. As more administrators and home users update their machines with the latest security updates and patches for their browsers, as well as the ability to automate the process, it becomes harder to use web browsers as an attack vector. However, although they may be updating their browser software, it is also true that many people forget to update third party add-ons like the Flash player [8]. In 2009 a number of malware "in the wild" (out there on the Internet) have exploited the PDF file format, Adobe Acrobat, Flash, a number of ActiveX components and Java. These third party add-ons are used to push users to other websites that have been compromised.

Downloads: While automating remote code execution is very attractive for attackers, there are many times when this level of sophistication is not required to compromise end-users' computers. In fact, some attacks still rely on endusers downloading executable files. To aid attackers, legitimate websites, some of which are high profile, are being compromised. As soon as these sites become infected, they can start serving malware, thus exploiting a user's tendency to trust content based on reputation. When a legitimate news site asks end-users to download an executable file (e.g., codec) to view an intriguing video, many will comply because they trust that website. Malware creators use a variety of techniques to convince users to visit poisoned sites, search results and download executables. are told to either download software to continue or else malware is downloaded while they are looking at the content.

Hybrid attack: While the web offers much greater scope for attackers, email still remains a powerful tool. Combined with the web, the threats not only multiply but the risk that the user becomes a willing prey is very high. One common trick is to use current news events to spread malware spam. Emails purporting to offer exclusive news, videos or files are popular online traps to open dangerous attachments or be redirected to infected or fake websites.

Malware infections cause a number of problems. Machines become unresponsive or sluggish resulting in users becoming frustrated and administrators spending precious time trying to find the problem. When a machine is infected, some administrators often want to simply re-install the operating system, however a responsible system administrator or security analyst would want to investigate and assess the situation before doing anything else. All of these tasks take time and resources. People have to stop working, the hardware has to be replaced and so on. Additionally, some malware creates a denial of service by design, increasing the possibility of an attack on the organization's infrastructure.

While most organizations understand denial of service very well – since it impacts productivity – many ignore the impact on confidentiality and integrity. Attackers are known to harvest sensitive information from compromised computers to carry out further and deeper attacks within the network. If they access the organization's data they can use this to sell to third parties and make a profit. Modern malware can create an automated process to harvest information from a network that has been breached. Once an attacker is on the inside, his or her work is significantly easier since on most networks, systems on the inside are trusted. This is what makes attacking web visitors through infected websites so attractive to the bad guys. End-users and their web browsers are already on the internal network. Unlike traditional network-based attacks, the victim connects to the attacker instead of the other way round. Even today, most defenses are still focused on preventing attackers from trying to connect to the victim, i.e., protecting the perimeter. Therefore the prime need is of the following:

Security Policies

Education alone is not enough. Organizations need security and user policies that can be enforced. These policies need to be reasonable and allow employees to do their job yet limiting actions that could be a security risk. This is easier said than done because many security policies and solutions impact usability. Therefore a good security analyst has a tough job finding a balance between security and helping employees to deliver and be productive. When policies are too strong, employees will find ways around the policies or become less productive – a situation that is untenable and unacceptable for a business [9]. Security policies are important but only effective when they are enforced and users are aware of them. It is highly recommended that businesses create acceptable user policies that every employee has to sign. Enforcement, however, is another story and requires more than just an employee's signature on a piece of paper that they will probably never see again. Technology is key here because it allows administrators to enforce policies across the network with minimal effort.

Using Technology

Although administrators are comfortable working with technology, they are somewhat limited in dealing with human nature. Policies and education are important but you will still find people who don't really care. If they want to open a file or click on a link they will do so. So administrators need to boost their arsenal with technology. Anti-spam and anti-phishing software, for example, will reduce the volume of unwanted email reaching the end-user, thereby reducing the risk of exposure to phishing scams and redirects in email. An administrator cannot (and should not) depend on end-users to follow policies to the letter. Bandwidth monitoring is one of the basic methods used to detect unwanted web traffic[10.] If, on average, an employee only browses a few websites every day, but this behaviour changes drastically, the administrator needs to look into the matter. The employees may have changed his or her browsing habits but it may also be a sign that the machine has been compromised. Bandwidth monitoring is an important tool for network and security administrators.

Content Filtering

While bandwidth monitoring gives an overall view of what is happening, it does not provide an in-depth analysis. Content filtering solutions for the web allow administrators to choose web content by file type and by location. Since certain file types available on the web are used more often by the bad guys, content filtering by file type is a very effective way to protect web browsers without impacting negatively on usability. This type of solution, for example, could be used to block web content that may execute on the client, such as exe files or installation files (MSI). Content filtering does not solve all security problems associated with web clients. In fact, malicious attackers make use of file types or web locations that are more often than not also used for legitimate purposes. For example, HTML does not typically include any malicious content. By embedding exploit code in HTML files, attackers typically bypass most content filtering that relies on blocking specific file types. Therefore having an antivirus solution – preferably deploying multiple antivirus engines – is still very effective, especially if that antivirus solution is good at catching this type of malicious content. Implementing an antivirus solution at a strategic point such as a web gateway or proxy has various advantages. It can be centrally managed and is separate from the end-user's computer that may be already infected. Prevention is only part of a holistic solution that should be in place to counter web-based threats. Regular monitoring and auditing will help detect security incidents that may have occurred. One reason why some security incidents are more serious than they should be is that these incidents are not handled correctly. In the case of web threats, an incident response plan typically covers things like roles and responsibilities as well as procedures on how to respond to incidents with guidelines on preparation, identification, containment and recovery.

Conclusion

It is concluded that the phenomenon of World Wide Web which is called as W3C now a days. Really this research has yielded the result that multimedia has made a crucial role in the dynamism of information technology which has revolutionised the whole world on the very same platform but some constituents are supporting it to be close to the modernisation. Many of the documents are the work of the World Wide Web Consortium (W3C), headed by Berners-Lee, but some are produced by the Internet Engineering Task Force (IETF) and other organizations. Usually, when web standards are discussed, the following publications are seen as foundational: Recommendations for markup languages, especially HTML and XHTML, from the W3C. These define the structure and interpretation of hypertext documents, Recommendations for stylesheets, especially CSS, from the W3C, Standards for ECMAScript (usually in the form of JavaScript), from Ecma International, Recommendations for the Document Object Model, from W3C. Security threats to web sites and web applications (webapps) come in many forms. Data centres and other assets used for hosting web sites and their associated systems need to be protected from all types of threat.

References

1. NCSA Mosaic — September 10, 1993 Demo. Totic.org. http://totic.org/nscp/demodoc/demo.html. Retrieved July 27 **(2012)**
2. Vice President Al Gore's ENIAC Anniversary Speech, Cs.washington.edu. February 14, 1996. http://cs.washington.edu/homes/lazowska/faculty.lecture/innovation/gore.html. Retrieved July 27, **(2009)**
3. Internet legal definition of Internet, West's Encyclopedia of American Law, edition 2, Free Online Law Dictionary, July 15, 2009. http://legal-dictionary.thefreedictionary.com/Internet, Retrieved November 25, **(2008)**
4. WWW (World Wide Web) Definition, TechTerms, http://techterms.com/definition/www. Retrieved february 19 **(2010)**
5. The W3C Technology Stack, World Wide Web Consortium. http://www.w3.org/Consortium/technology. Retrieved April 21, **(2009)**
6. Hamilton Naomi, (July 31, 2008), The A-Z of Programming Languages: JavaScript, Computerworld, IDG, http://computerworld.com.au/article/255293/-z_programming_languages_javascript. Retrieved May 12, **(2009)**
7. Buntin Seth, (23 September 2008), jQuery Polling plugin, http://buntin.org/2008/sep/23/jquery-polling-plugin/, Retrieved 2009-08-22 **(2009)**
8. Berners-Lee, Tim, Frequently asked questions by the Press, W3C, http://w3.org/People/Berners-Lee/FAQ.html, Retrieved July 27, **(2009)**
9. automatically adding www.___.com, mozillaZine. May 16, 2003. http://forums.mozillazine.org/viewtopic.php?f=9&t=10980. Retrieved May 27, 2009.
10. Masnick Mike, (July 7, 2008), Microsoft Patents Adding 'www.' And '.com' To Text, Techdirt, http://techdirt.com/articles/20080626/0203581527.shtml, Retrieved May 27, **(2009)**