Revista Publicando, 2(5). 2015, 65-77. ISSN 1390-9304

Niveles de Impacto: Heartbleed Bugs vs. Bash Bugs

Janeth Inés Mora Secaira[1], Juan Pablo Verrezueta Vásquez[2]

1 Universidad Técnica Estatal de Quevedo, jmora@uteq.edu.ec

2 Distribuidora Luber, juanpabloberrezueta@gmail.com

RESUMEN

Las relaciones sociales, económicas y culturales dependen, cada vez más, de las tecnologías e infraestructuras de la información y comunicación (ciberespacio), haciendo necesario articular un sistema nacional de seguridad la ciberseguridad (Fojon, 2010). Las vulnerabilidades de los sistemas de información sumada a la diversidad de explotarlas en el segundo trimestre del año 2014 dejaron expuesto por más de dos años desde su verificación el agujero de Seguridad o Heartbleed Bug presente en la librería OpenSSL (Gujrathi, 2014), la cual permitía la lectura de cierta información presente en la RAM del computador, en escenarios a nivel empresarial del tipo financiero, proveedores de servicios electrónicos entre otros, comprometiendo información de millones de usuarios. Hasta la fecha de su rectificación Heartbleed, fue considerado el principal error de las fallas que han sufrido los servidores a lo largo de la historia, sin ser más que un error de programación, que de a poco fue solucionado mediante la implementación de parches y actualizaciones a las librerías del protocolo OpenSSL. Lamentablemente, para septiembre del año 2014, se determinó la presencia de una nueva vulnerabilidad Bash, la misma que se basa en la línea de comandos que se ejecuta en plataformas Linux y Mac OS, remitiéndose a un proceso simple como el cortar y pegar parte del código.

Para evidenciar estas vulnerabilidades se plantea un escenario virtual. De esta forma se determinará el nivel de impacto que presenta cada una de estas problemáticas, realizar una comparación entre si e indicar cuales serían las medidas a tomarse en caso de presentarse dichas problemáticas en estas soluciones informáticas.

Revista Publicando, 2(5). 2015, 65-77. ISSN 1390-9304

Palabras claves: Heartbleed, Bash, bug, Ciberseguridad, OpenSSL, Heartbeat, payload, Handshake, Vulnerabilidad.

Revista Publicando, 2(5). 2015, 65-77. ISSN 1390-9304

Impact levels : Heartbleed vs. Bugs Bash Bugs

ABSTRACT

Social, economic and cultural relations depend, increasingly, technologies and infrastructure of information and communication (cyberspace), making it necessary to articulate a national security system cybersecurity (Fojon, 2010). The vulnerabilities of information systems coupled with the diversity of exploit in the second quarter of 2014 left exposed for more than two years since its verification or heartbleed security hole present in the OpenSSL Bug (Gujrathi, 2014) library, which allowing reading some information in the RAM of the computer, enterprise-level scenarios of financial, electronic service providers and others, compromising information from millions of users. Until the date of rectification heartbleed, was considered the main mistake of the flaws that have suffered servers throughout history, without being more than a programming error, that little by little was solved by deploying patches and updates OpenSSL libraries to protocol. Unfortunately, in September 2014, the presence of a new vulnerability was determined Bash, the same that is based on the command line that runs on Linux and Mac OS platforms, referring to a simple process as the cutting and pasting of the code.

To demonstrate these vulnerabilities a virtual scenario arises. Thus the level of impact that has each of these issues will be determined, a comparison between them and indicate what would be the measures to take in case of such problems in these solutions.

Keywords: Heartbleed , Bash , bug , Cybersecurity , OpenSSL , Heartbeat , payload , Handshake , Vulnerability.

Revista Publicando, 2(5). 2015, 65-77. ISSN 1390-9304

1. INTRODUCCIÓN

Con la idea de concientizar cada día más a nuestra sociedad en la importancia de la seguridad informática y dar a conocer los riesgos que ciertas vulnerabilidades han marcado hitos importantes en este campo. Entre estas hemos decidido citar dos bugs: El Heartbleed (Durumeric, 2014) y Bash (Saleem, 2015); en el momento de la divulgación del primero, alrededor del 17% de los sitios web seguros de todo el mundo decía ser vulnerables al error, tomando en cuenta las implementaciones de OpenSSL (Ramos, 2015) están presentes en servidores Apache, Microsoft y nginx.

Apache continúa siendo el servidor web dominante con cerca del 40% del mercado con más de la mitad de sitios activos de internet funcionando en él, seguido por Microsoft con el 28% y nginx tiene un porcentaje del 14%, según lo citado por Netcraft (Netcraft, 2015) a marzo del presente año.

Índice de penetración de servidores por desarrolladores a marzo 2015

Developer	February 2015	Percent	March 2015	Percent	Change
Apache	342,480,920	38.77%	337,175,536	38.39%	-0.38
Microsoft	253,484,221	28.69%	245,496,533	27.95%	-0.74
Nginx	130,093,899	14.73%	127,191,696	14.48%	-0.25
Google	20,238,057	2.29%	20,097,702	2.29%	-0.00

Tendencias servidores web (Netcraft, 2015)

De estas últimas estadísticas podemos señalar que la tendencia se mantiene considerando que cerca del 80% de los servidores implementados a nivel mundial presentaban dicha vulnerabilidad por dependencia directa a la librería mencionada. El error Heartbleed es en aplicación de OpenSSL de la extensión TLS/DTLS descrito en la RFC-6520 (Internet Engineering Task Force - IETF, 2012).

Revista Publicando, 2(5). 2015, 65-77. ISSN 1390-9304

Cuando se es explotado conduce a la fuga del contenido de la memoria del servidor para el cliente y el cliente al servidor. Sin embargo, este fallo ha dejado gran cantidad de claves privadas y otros secretos expuestos a Internet. Teniendo en cuenta la larga exposición, facilidad de explotación y ataques sin dejar rastro esta exposición debe ser tomado en serio, la afectación de Heartbleed considera que OpenSSL es la biblioteca criptográfica de código abierto más popular e implementación de TLS utilizada para cifrar el tráfico en Internet, es propenso a ser afectado directa o indirectamente.

2. METODOS

Esta investigación se define como de tipo exploratoria y descriptiva, por tratarse de una investigación que pretende describir la situación en la que se encuentran las vulnerabilidades de los Heartbleed y Bash Bugs, para esto el uso de entornos virtuales cada día simplifica la posibilidad de implementar soluciones con fines investigativos (Britos, Vargas, Arias, Giraudo, & Veneranda, 2013). El estudio exploratorio a seguirse permite aproximarnos a una realidad poco conocida, como es el uso de los sistemas computacionales y los servicios que ella involucra, por parte de las los usuarios finales, a quien nosotros representamos.

A. HEARTBLEED BUGS Y BASH BUGS

Un error en una librería de software de código abierto llamado OpenSSL que está diseñado para cifrar las comunicaciones entre la computadora de un usuario y un servidor web, que permite mostrar a un atacante hasta 64 kB de memoria a un cliente o servidor conectado, rellenando el paquete con datos de la memoria RAM, la cual fue denominada Heartbleed ya que afecta a una extensión de SSL conocido como Heartbleed. Oficialmente conocida como CVE-2014-0160 (CVE-2014- 0160, 2015), el error ha estado presente desde hace unos años y a finales del primer trimestre de 2012 la falla de seguridad en dicho protocolo se determinó que la vulnerabilidad permite a los hackers robar con facilidad sustraer parte de datos previamente seguros. Ésta referencia es el estándar para los nombres de vulnerabilidad de seguridad información mantenida por MITRE (MITRE, 2015).

Revista Publicando, 2(5). 2015, 65-77. ISSN 1390-9304

De su parte, a inicios del cuarto trimestre del año 2014 (VIESTINTAVIRASTO, 2014), teniendo como referente las vulnerabilidades en las comunicaciones encriptadas se confirma la falencia en el código CVE-2014-6271 (CVE-2014- 6271, 2014) presentando una nueva amenaza con la que un atacante puede brindar variables de entorno especialmente diseñadas que contienen comandos arbitrarios que se ejecutarán en los sistemas vulnerables bajo ciertas condiciones las cuales han sido denominadas como vulnerabilidades de inyección código Bash y se encuentran especificadas en CVE-2014-7169 (CVE, 2014).

B. PROTOCOLO SSL/TLS Y ATAQUES

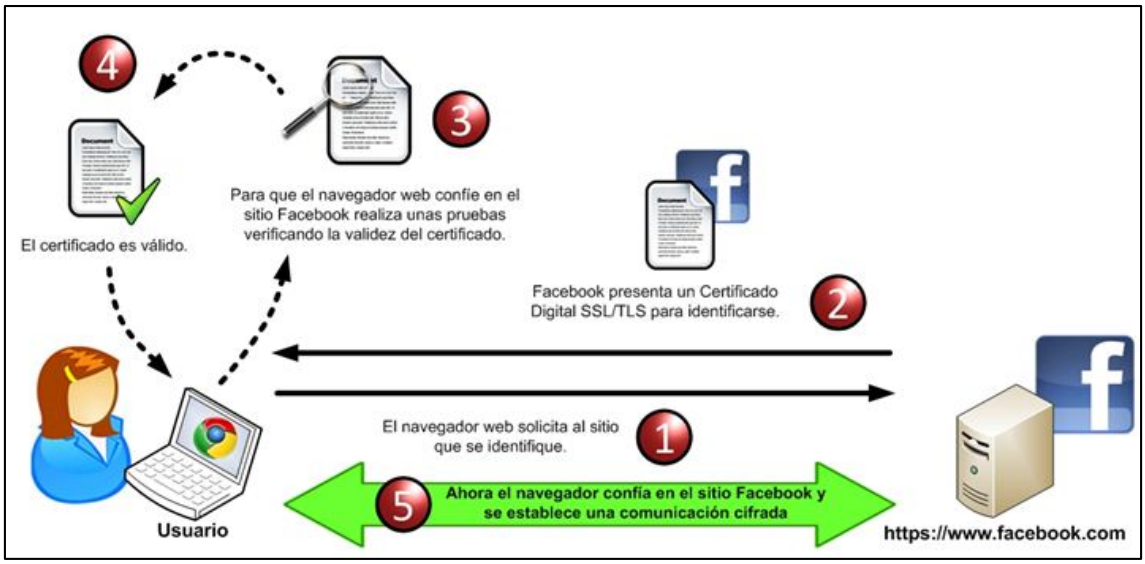
SSL/TLS (Jawi, 2015). Los protocolos que hace uso de certificados digitales para establecer comunicaciones seguras a través de Internet (Ramírez López & Espinosa Madrigal, 2011).

Recientemente ha sido sustituido por TLS (Amour, 2015) el cual está basado en SSL y son totalmente compatibles. Intentar acceder al sitio web ejemplo Facebook de forma segura, es decir, usando “https” en la dirección web. Inmediatamente, aparecerá la página en pantalla y en alguna parte del navegador se observará un “candado”, dependiendo del navegador que use.

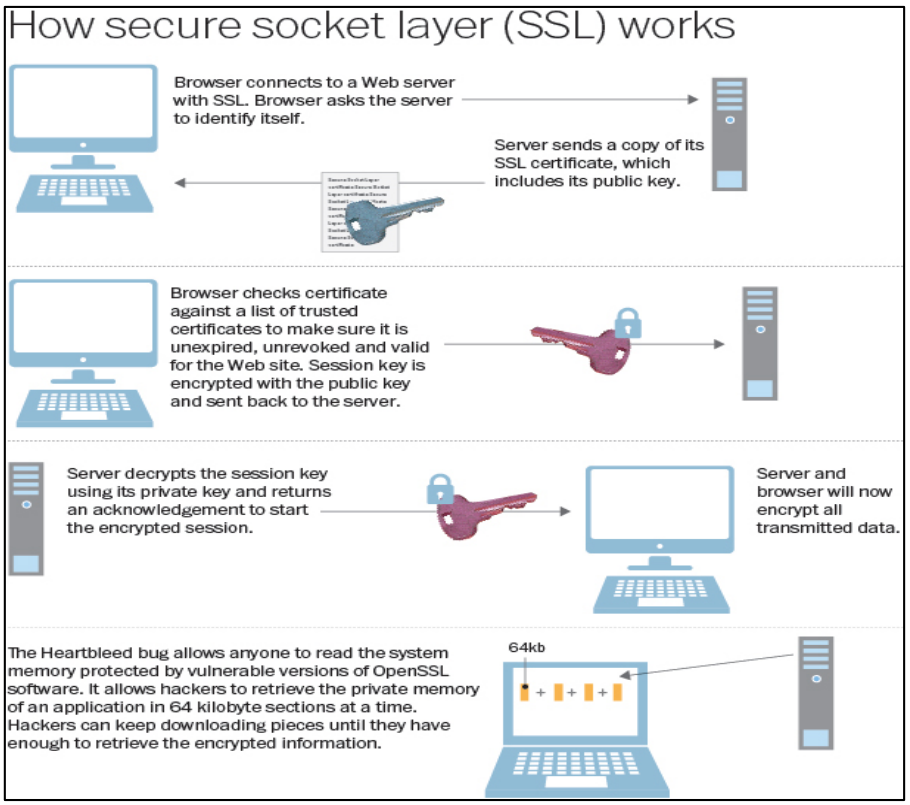
Cuando se conecta con un sitio de web seguro, hay una conexión para configurar la sesión segura. Su navegador solicita y verifica el certificado del sitio, genera una clave de cifrado para la sesión segura y encripta con la clave pública del sitio, y el sitio descifra utilizando la clave privada correspondiente comenzando la sesión. El navegador solicita datos desde el sitio, este último devuelve datos hasta la siguiente solicitud. (Fernández, 2012).

Sin embargo es útil para ambos lados de una conexión segura para asegurarse de que el otro aún está activo. En la siguiente figura (Blacksun, 2014) se presenta el procedimiento en cual se presenta la vulnerabilidad en una comunicación.

Revista Publicando, 2(5). 2015, 65-77. ISSN 1390-9304



Funcionamiento general de SSL/TLS



Acción del Heartbleed (Blacksun, 2014)

Revista Publicando, 2(5). 2015, 65-77. ISSN 1390-9304

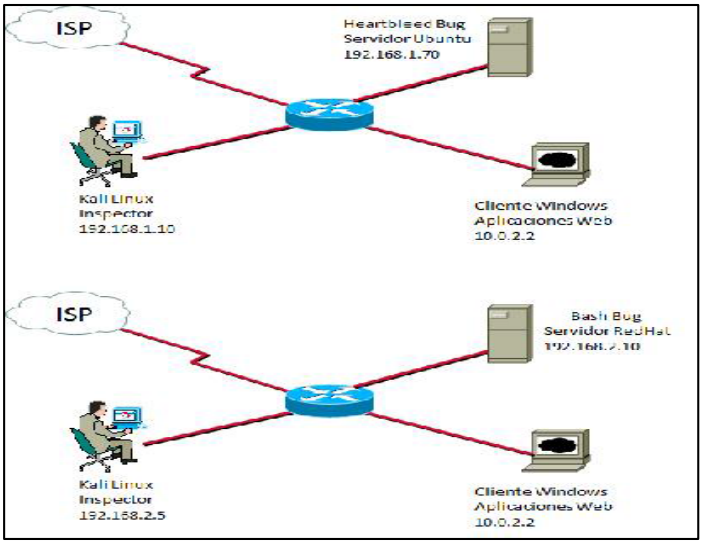
El atacante tendrá que escoger buscando patrones. Pero potencialmente nada en absoluto podría ser capturado, incluyendo las claves de encriptación, credenciales y mucho más. Puesto que la explotación de este fallo no deja ningún rastro, realmente no podemos decir cuántos datos supuestamente seguro ha sido robado. Pasará este tema específico pero destacar una característica importante del mundo conectado e ilustra la necesidad de las empresas y los proveedores de seguridad a ambos ser ágiles en cómo abordar cuestiones como éstas y adoptar técnicas de inteligencia.

Todos los productos que utilizan el Shell Bash se ven afectados por esta vulnerabilidad por lo que es preciso analizar los valores de las variables de entorno, lo que es presenta un alto grado de peligrosidad ya que existen muchas maneras posibles para que un bash puede ser llamado por una aplicación.

Muy a menudo, si una aplicación se ejecuta otro código binario, Bash es invocado para ello. Debido a la extendida utilización de la shell Bash, este tema es muy grave y debe ser tratada como tal (ZNET, 2014).

3. RESULTADOS

Para el análisis en el presente proyecto en cuanto hace referencia al Heartbleed se ha implementado un escenario virtual en Oracle VM VirtualBox simple, el mismo que podrá determinar si nuestro servidor y cliente están expuestos o no a ésta vulnerabilidad.



Revista Publicando, 2(5). 2015, 65-77. ISSN 1390-9304

Modelos de Evaluación Haeartbleed y Bash Bugs

Mediante técnicas de Penetration Test, a través del uso del comando nmap (NACIONAL, 2012) detectamos una función, vulnerabilidades, mediante una sintaxis como la siguiente:

Servidor Ubuntu:

Nmap –sV --script=ssl-heartbleed

192.168.1.70

Cliente Windows:

Nmap –sV --script=ssl-heartbleed

10.0.2.2

A más de ello podemos utilizar las referencias de acuerdo a la versión de la librería instalada de openssl, utilizando el comando openssl version. Y si la máquina objetivo es vulnerable se registrará.

En este caso se colocó una bomba fork, como ejemplo sencillo dentro de una función, obteniéndose el colapso del equipo al realizar el consumo excesivo de recursos de memoria en el servidor, para lo cual se generó un script que se detalla a continuación.

```
#!/bin/bash

clear

echo “Ataque DoS”

date > h_ini.txt

sar –u 2 4 >> datos

bomba()
```

En donde la función bomba está dada de la siguiente forma:

```
Bomba()

{bomba | bomba&};
```

Revista Publicando, 2(5). 2015, 65-77. ISSN 1390-9304

```
bomba
```

Dicho script permite determinar el uso recursivo de la memoria del sistema hasta bloquear totalmente el equipo. Entre las grandes funciones afectadas se tienen denegación de servicio (DDoS), IRC bot, funciones que intenta adivinar las contraseñas y los inicios de servidores vulnerables para lo que utilizan listas de contraseñas débiles como root, admin, usuario, login, y 123456 muy comunes en etapas de levantamiento de soluciones y que de no tener una cultura de seguridad informática, traerán un sin número de problemas.

De acuerdo a los datos expuestos en un análisis (Graham, 2014) analizando las vulnerabilidades del puerto 80 al menos 3.000 sistemas son vulnerables a este tipo de errores, manteniéndose alerta en servicios DHCP y pasando desapercibidos por firewalls e infectando diversidad de sistemas.

Las dos vulnerabilidades analizadas presentan una característica común que es la autenticación y denegación de servicio a los usuarios. Sin embargo es preciso analizar por separado en primera instancia los servidores y luego a nivel de cliente las soluciones a implementarse. En el caso de los heartbleed bugs los servidores deberán tomarse en forma primordial la revocatoria de los certificados digitales del servidor, proceder a instalar parches o nuevas versiones de OpenSSL, instalar nuevos certificados digitales y finalmente proceder a solicitar a todos los usuarios registrados al servidor el cambio inminente de las contraseñas. Mientras que usuarios o clientes de servidores, deben realizar cambiar todas las contraseñas que usan a las distintas plataformas.

Por otra parte en cuanto hace referencia a los Bash Bugs a nivel de servidores, estos requieren de la implementación de parches o actualizaciones a los sistemas operativos, Herramientas de detección de Shellshock y soluciones en hardware y software para monitorear la actividad de la red.

Los usuarios finales deberán realizar las actualizaciones de seguridad en cada uno de sus sistemas operativos y como herramienta adicional aplicativos para la detección de shellshock.

Revista Publicando, 2(5). 2015, 65-77. ISSN 1390-9304

Heartbleed presentó un impacto considerable al extenderse por distribuciones como Debian, Ubuntu, CentOS, Fedora 18, OpenBSD, OpenSuse, las mismas que ya han presentado actualizaciones y parches para minimizar las oportunidades de afección por esta vulnerabilidad, sin embargo, el efecto se sigue expandiendo puesto que todas las comunicaciones han ido de a poco utilizando SSL con la librería openSSL, por lo que se debe analizar constantemente la red. En lo que respecta a los Bash Bugs, teniendo en cuenta su presencia por más de 25 años puesto que es parte fundamental del kernel, en plataformas Unix, Linux, MacOS al ser base fundamental de éstas, ya tienen algunas soluciones que en forma conjunta pueden afectar a distribuciones como Debian, Ubuntu, Red Hat, CentOS, Novell/SUSE.

4. CONCLUSIONES

Los administradores de la red deben mantenerse en constante evaluación de sus plataformas, de tal forma que conozcan más sobre el correcto funcionamiento y fallos que este puedan presentar. En Ecuador tomando en cuenta las cifras expuestas a nivel mundial y su tendencia a la implementación de soluciones open source nos obliga a concientizar que se propone a ser atacados, para lo que debemos seguir las recomendaciones que plantean los desarrolladores de nuestras soluciones informáticas (parches y actualizaciones de software) para garantizar su estabilidad y seguridad.

HeartBleed es una vulnerabilidad conocida, explotada de manera habitual por todas las herramientas, pero seguro que seguirá presente en múltiples sitios tomando en cuenta el incremento de servicios que a los usuarios nos pueden ofrecer. Es así que realizar escaneos completos de todos los equipos y dispositivos de acceso a la de la red, evaluando todos los puertos sería una buena práctica para localizar librerías OpenSSL que mantengan vulnerabilidades. Así mismo a nivel empresarial establecer políticas de seguridad informática nos ayudarán a mantener la integridad de la información que manejan nuestros equipos, es así que utilizar la práctica de cambio de password es en forma periódica minimizará la posibilidad de tener acceso a cuentas de los usuarios de parte atacantes.

Revista Publicando, 2(5). 2015, 65-77. ISSN 1390-9304

De su parte, al actuar Bash como un intérprete de lenguaje de comandos permite que el usuario escriba los comandos en una ventana en texto simple permitiendo realizar una serie de instrucciones o ejecutar comandos que se le pasa por las aplicaciones en caso de tener Bash Bugs en la plataforma de servicios. Lo crítico está en el acceso a las variables de entorno que afectan a los procesos que se ejecutan en un equipo. Y dado que es un ambiente nativo de trabajo del sistema, tendremos acceso a una serie librerías que al ser manejadas por agentes externo podrían repercutir negativamente en múltiples campos de acción, presentando escenarios en los que se caractericen por simples pérdidas de información hasta la totalidad de la misma, o colapso total de nuestra red empresarial y de sus servicios, de donde se desprende la verdadera necesidad de gestionar mecanismos, entendiéndose hardware y software, para realizar el monitoreo completo de los elementos de la red, puesto que únicamente de esta forma se podrá minimizar la posibilidad de vernos afectados por esta vulnerabilidad.

5. REFERENCIAS BIBLIOGRÁFICAS

Amour, L. S., & Petullo, W. M. (2015). Improving Application Security Through TLS-Library Redesign

Britos, D., Vargas, L., Arias, S., Giraudo, N., & Veneranda, G. (2013). Laboratorio Remoto Virtual para la Enseñanza de. Tercera Conferencia de Directores de Tecnología de Información y Comunicación en las Instituciones de Educación Superior: soluciones de Enseñanza y la Investigación. Cartagena de Indias.

Blacksun, S. (2014). *BLACKSUN*. Obtenido de HeartBleed Explained: https://support.blacksun.ca/index .php?/News/NewsItem/View/74/ blacksun-update--heartbleedexplained

Cve-2014-0160. (2015). *common vulnerabilities and exposures*. obtenido de

https://cve.mitre.org/cgibin/ cvename.cgi?name=CVE- 2014-0160.

Cve-2014-6271. (2014). *common vulnerabilities and exposures*. obtenido de

https://cve.mitre.org/cgibin/ cvename.cgi?name=cve- 2014-6271

Durumeric, Z., Kasten, J., Adrian, D., Halderman, J. A., Bailey, M., Li, F., ... & Paxson, V. (2014, November). The matter of Heartbleed. In *Proceedings of the 2014 Conference on Internet Measurement Conference* (pp. 475-488). ACM.

Revista Publicando, 2(5). 2015, 65-77. ISSN 1390-9304

Fernández, V. (2012). *Dolthink Tecnología e Inovación.* Obtenido de Compras seguras en Internet: http://www.dolthink.com/compr as-seguras-internet.html

Fojón Chamorro, e., & sanz villalba, á. f. (2010). ciberseguridad en España: una propuesta para su gestión. ARI, 102, 2010.

Graham, R. (2014). *Bash 'shellshock' bug is wormable*. Obtenido de http://blog.erratasec.com/2014/0 9/bash-shellshock-bug-iswormable.

html#.VRhefuGRZoh Internet Engineering Task Force - IETF. (2012). Transport Layer Security (TLS) and Datagram Transport

Jawi, S. M., Ali, F. H. M., & Zulkipli, N. H. N. (2015). Nonintrusive SSL/TLS Proxy with JSON-Based Policy. In *Information Science and Applications* (pp. 431-438). Springer Berlin Heidelberg.

Nacional, c. c. (2012). guía avanzada nmap. Valencia

Netcraft. (2015). March 2015 Web Server Survey. Obtenido de http://news.netcraft.com/archives/2015/03/19/march-2015-web-server-survey.html#more-18769

Ramos, D. A., & Engler, D. (2015). Under-Constrained Symbolic Execution: Correctness Checking for Real Code. In 24th USENIX Security Symposium (USENIX Security 15). USENIX Association.

Saleem, S. (2015). Analysing the Resolution of Security Bugs in Software Maintenance (Doctoral dissertation, The Open University).

Servidio, j. s., & taylor, r. d. (2015). safe and sound: cybersecurity for community banks. journal of taxation & regulation of financial institutions,28(4).

ZNET. (2014). First attacks using'shellshock' Bash bug discovered