A Review and Analysis on Heartbleed on Italian Websites, a Year Later

Vito Santarcangelo[1,2] Fabrizio Valenti[2] Muhammad Imran Tariq[3] Claudio Fornaro[4]
Giuseppe Oddo[1,2]
Domenico Di Carlo Jr.[1]

[1]Centro Studi S.r.l.
Zona Industriale
Buccino, Italia

[2]iInformatica S.r.l.s.
Corso Italia,77
Trapani, Italia

[3]Dept. of Info. Technology
Superior University Lahore
Lahore, Pakistan

[4]Universitá Telematica Uninettuno
Corso V. Emanuele II
Roma

***Abstract*—Heartbleed, a big Open Secure Socket Layer (OpenSSL) vulnerability appeared on the web on 7th April 2014. This highly risked vulnerability enabled attackers to remotely read protected memory contents from Hyper Text Transfer Protocol Secure (HTTPS) sites. In this paper, the authors will review and analyze Heartbleed vulnerability effects on secured websites, a year later (April 2015). To accomplish this, we conducted an analysis on a dataset of 100 Italian public and private sector websites like banks, stock exchanges, Cloud Organizations and services on HTTPS websites, thereby obtained that only 1% of the websites show the vulnerability. However, new vulnerabilities as Padding Oracle on Downgraded Legacy Encryption (POODLE) & Factoring Attack on RSA-Export Keys (FREAK) affect a lot of websites, particularly the websites used as point of accesses of Italian telematics process. We concluded the paper with the analysis of the Cloud risks that are very harmful for the Cloud customers as well as the Cloud venders due to Heartbleed attack.**

Keywords–Heartbleed; OpenSSL; Poodle; Freak; Vulnerability.

I. INTRODUCTION

The Cyber Security is living an awkward moment caused by transition from the technical community to the public one [3]. Cyber Security is one of the most important topic in our days, because people which interact with the external world have to navigate in a secure mode. Heartbleed is a famous security bug spread in April 2014 [6]. It has caused a serious vulnerability bug in an open source cryptographic software library called OpenSSL. This library implements Transport Layer Security (TLS), an upgrade of Secure Socket Layer (SSL), a protocol developed to guarantee and to provide communication security between two or more devices over the Transmission Control Protocol (TCP) networks. This bug has involved the commercial transactions on the Internet of hundred million people around the world. It allowed everyone to listen to secure traffic exchanged between endusers. In this way, attackers steal the traffic (such as access credentials, passwords, payment cards of the users) by using secret keys used for traffic encryption [1]. It is considered one of the devastating disasters occurred in the internet age. This bug is known as Heartbleed because it uses the Transport Layer Security (TLS) protocol heartbeat extension; when it is broken, the secure communication channel between server and client is altered giving attackers the possibility to gather data. Heartbleed is the most famous cyber-attack in the last years.

A. Transport Layer Security (TLS)

The Transport Layer Security (TLS) is a protocol specified by Internet Engineering Task Force (IETF) as an enhancement over Netscapes Secure Socket Layer in 1999 [4]. It manages encryption and authentication on the TCP networks. Its peculiarity is enforcing security and data integrity. TLS protocol uses some kind of algorithms called ciphers to guarantee data integrity. TLS, in order to create a secure channel communication, uses a Public Key Infrastructure (PKI). As a consequence, the remote peers exchange information using an asymmetric cryptography mode. Each peer has two different keys: a private key and a public key. The first one is used by senders to sign digitally messages through a digest. The second one is used by receivers to validate the messages integrity. They create a new digest and then they compare it with the digest received message of the public key. If the two digests match, the message will be validated and verified. Two end peers can exchange information along the Transmission Control Protocol (TCP) networks using a public key called X.509 certificate. If an X.509 certificate is very weak it can be cracked by the attackers. For this reason, the system cryptography used to build certificates must be robust to prevent most attacks (such as Man in the Middle MITM). To decrease the weakness, the Operation Systems need robust key random generators. The longer the key, the more difficult is to break in. For example, a 1024 bits Rivest Shamir and Adelman (RSA) is more vulnerable than a 2048 bits RSA. If attackers obtain the private key, they can listen TLS traffic and decrypt it. There are several methods useful to prevent attacks. For example, there are different alternatives to the RSA keys, such as Digital Signature Algorithm (DSA) and Elliptic Curve Digital Signature Algorithm (ECDSA) [4]. Meanwhile, this can be mitigated using Perfect Forward Secrecy (PFS), a property ensuring a security transmission so that if a long-term key is compromised, the session key derived from it is safeguarded. Therefore Heartbleed is an important implementation vulnerability different from other attacks, such as Crime, Beast and Breach [12]. In this case, a programming issue occurred in Open SSL which generated implementation vulnerability in Transport Layer Security (TLS) protocol heartbeat extension [7].

B. Cloud Computing

Cloud computing means to hire the services of cloud vender on pay as per use basis over the internet [15]. The

 ISBN: 978-1-61208-427-5

Cloud customer uses Web Browser to access the services that it rendered from Cloud vender. Although immense research has been carried out to find out the security challenges and issues on the Cloud but Cloud is not adequately secured as traditional IT computing. Secure connection between Cloud customer and Cloud vender is highly important which must be secured by the implementation of encryption and TLS/SSL [13]. The Cloud customer store and processes its highly sensitive data on the machines that it rendered over the Cloud network. All the data travels on internet from customer computer to Cloud Service Provider (CSP). There are lot of free cloud services available over the internet like a Dropbox, Google Docs, Flicker and etc. Most of the Cloud organizations use Open Secure Sockets Layer (OpenSSL) to provide a secure platform for transformation of data over the internet. Among Cloud security risks, the Heartbleed security vulnerability made attackers to breakdown Cloud encrypted communication by exploiting serious security vulnerability in OpenSSL library and get about 24-55% protected memory contents of the Cloud machines and ultimately attacker become successful to get significant amount of information. The Cloud organizations which have customers running TLS and not relying upon OpenSSL are not infected with Heartbleed vulnerability, but its ratio is very low. The research found a number of Cloud risks that are very harmful for the Cloud Websites. The Cloud risks are given in the Section V of this research paper. The authors keeping in view the Cloud risks, checked the Cloud websites of the Italy. The list of the websites that are infected due Heartbleed bug is not appended in the paper due to security concerns of these websites.

C. New vulnerabilities

Heartbleed is a 'server side' attack, therefore it can be conducted by a remotely attacker simply knowing the public IP address of the target. In this work we show also new web vulnerabilities as POODLE and FREAK. These vulnerabilities are 'Man in the Middle' (MIMT), therefore, their impact is only for clients and local network environment ('client side'). POODLE is a vulnerability of clients (e.g. web browsers) for the support of SSL 3.0 [11]. It allows an attacker to decipher 'SECURE' HTTP cookies on the local network. FREAK allows an attacker to intercept HTTPS connections (MIMT) between vulnerable clients and servers and force them to use weakened encryption[16]. At the website 'freakattack.com' it is possible to control the vulnerability of web-browsers and of a great dataset of vulnerable websites (servers) to FREAK.

The paper is organized as follows: in Section 2, we describe the heartbeat vulnerability and the relative Heartbleed attack. Section 3 presents a simulation of Heartbleed attack through Python Script Language. Section 4 present an own HTTPS Italian website dataset developed for this paper and the relative results obtained using Qualys SSL Labs online tool. Section 5 shows a detailed analysis about Cloud Risks of Heartbeat vulnerability. Finally, in Section 6 we discuss open challenges about HTTPS website security.

II. HEARTBEAT AND HEARTBLEED

Heartbleed attack is a bug present in OpenSSL versions 1.0.1 through 1.0.1f. It allowed attackers to steal and to analyze private cryptographic keys. The first reason that allowed this kind of attack is that there were not security checks in code that implemented TLS protocols. Attackers had access to memory space used by TLS to store data like session key, in the server. Therefore, attackers could handle traffic exchange from clients to server and vice versa, stealing password and other user's information. As indicated in [2], The affected versions of hardware/software were those which used vulnerable versions of OpenSSL. Some operating system distributions that have been shipped with potentially vulnerable OpenSSL version were Debian Wheezy (stable), OpenSSL 1.0.1e-2+deb7u4; Ubuntu 12.04.4 LTS, OpenSSL 1.0.1-4ubuntu5.11; CentOS 6.5, OpenSSL 1.0.1e-15; Fedora 18, OpenSSL 1.0.1e-4; Open BSD 5.3 (OpenSSL 1.0.1c 10 May 2012) and 5.4 (OpenSSL 1.0.1c 10 May 2012); FreeBSD 10.0 - OpenSSL 1.0.1e 11 Feb 2013; NetBSD 5.0.2 (OpenSSL 1.0.1e); Open SUSE 12.2 (OpenSSL 1.0.1c). It uses TLS protocol Heartbeat extension kept the channel communication alive when there were not information to exchange between end-users. For this reason this bug allowed stealing under normal condition all the information handled by the TLS encryption. The Heartbeat Message was exchanged using an SSL3 RECORD structure.
The Heartbleed mechanism consists of some phases[5]. The detail of these phases (Figure 1) is given below:
1. Potentially attackers, send any heartbeat messages request to device running a vulnerable version of OpenSSL. These simple message consists of two key fields: a payload length (64 KB) and data.
The structure is the following:

```
struct ssl3Record
{
int length;
char *data;
}
```

2. When the peer gets the message request, this is processed and the request is written to memory allocating a buffer for response;
3. OpenSSL copies the payload content into buffer allocated, without bounds checks, considered it trust;
4. OpenSSL returns a message response containing the original payload and other private information like long terms server private keys, session ticket keys, confidential data and TLS session keys.

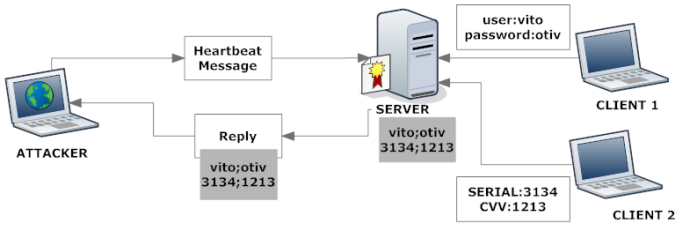

Figure 1. Heartbleed attack

Through this implementation vulnerability, the attackers were allowed to access data contained in the security infrastructure that used OpenSSL gathering personal information. Meanwhile it is not possible to know if someone had exploited this bug against our system architecture because this implementation vulnerability does not leave any trace in the logs. Today, there are different solutions adopted to face and to prevent this bug; the first is to update the OpenSSL version or alternatively, the OpenSSL code has to be recompiled remov-

 ISBN: 978-1-61208-427-5

ing from it the time option DOPENSSL_NO_HEARTBEATS.

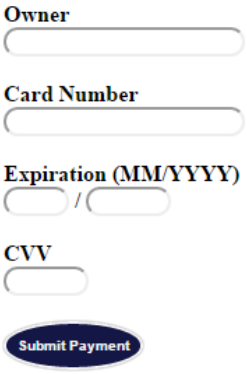

Figure 2. Payment Test Form

III. HEARTBLEED ATTACK SIMULATION

To understand the concrete impact of the Heartbleed, in this section, we show Heartbleed attack in a local environment. To conduct this attack, on Heartbleed vulnerability, we have considered a solution PYTHON based on the exploit code of eelsivart [15]. This solution considers a hostname as parameter and attacks the Heartbleed vulnerability. As case study we have considered a Windows Environment with XAMPP 1.8.3, APACHE 2.4.7, PHP 5.5.9 and OpenSSL/1.0.1e (known version vulnerable to Heartbleed). We have considered two PHP forms: a classic contact form and a classic payment one given in Figure 2 above. In this example, we can imagine a user that inserts information in two website forms protected by SSL. The output in Figure 3 shows the parameters of contact form extracted by Heartbleed attack. The parameters (name, email, subject, and message) of contatti.php are then visible to the attacker. The same output appears for the form credito.php that shows the credit card parameters inserted (Figure 4). Then, it is possible for an attacker to schedule distributed extractions to obtain a lot of information through Heartbleed. These extractions can be automated and elaborated by a simple text parser that catalogs only the relevant information. Consequently, this example underlines the dangerousness of Heartbleed attack that allows an attacker to retrieve a lot of useful information (e.g. usernames, passwords, credit cards).

```
C:\Python27>python.exe C:\testheartbleed.py 192.168.18.229

Connecting to: 192.168.18.229:443

HEARTBEAT WARNING: 192.168.18.229:443


User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/41.0.2272.101 Safari/537.36

Referer: https://127.0.0.1/form/contatti.php

nome=Domenico&email=domenico.dicarlo%40dicarlo.biz&oggetto=Informazioni+commerci
ali&messaggio=Salve%2C+vorrei+chiedere+informazioni+sul+prezzo+del+prodotto+n%C2
%B0535+presente+nel+vostro+listino.%0D%0ASaluti+Domenico&submit=Invia+messaggio|
....%.A.......d/+...6...6.8...8....P
```

Figure 3. Heartbleed test

```
C:\Python27>python.exe C:\testheartbleed.py 192.168.18.229

Connecting to: 192.168.18.229:443

HEARTBEAT WARNING: 192.168.18.229:443


User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/41.0.2272.101 Safari/537.36

Referer: https://127.0.0.1/form/credito.php

intestatario_carta=Vito+Santarcangelo&numero_carta=1717171717171717&mese=11&anno
=11&cvv=121&submit=Effettua+Pagamento..*mQ.;E..[...R,/+...6...6.8...8....P
```

Figure 4. Heartbleed test

IV. 2015 WEBSITE ANALYSIS

In this section we show the experiment conducted on an own dataset of Italian HTTPS website (.it extension). The dataset of websites has been built by us considering the result of the query ‘inurl:https:// .it’ on ‘www.google.it’ using the search operator ‘inurl:’, and categorizing the Italian websites in three groups: Bank, Institution, Other (e.g. Mail, Cloud, E-commerce, Bet Services, Billing). We have named this dataset as ‘IT-HTTPS-DB’. We can share it (by e-mail request at vito.santarcangelo@centrostudi.biz) for research purposes. Our database is made of 100 websites: 35 bank, 35 public institution and 30 other HTTPS websites.

We have conducted the vulnerability test with the use of SSL Labs online tool considering as metric their SSL Server Rating [9]. SSL Server Rating is a metric of Qualys SSL Labs that assigns a letter from A (strong) to F (weak) to evaluate the security level of a website.

The results obtained are that only the 1% is characterized by Heartbleed vulnerability, so the impact of this attack can be considered almost completely solved in only one year. Nevertheless, we can see that a lot of websites are affected by some important Man in the Middle (MIMT) vulnerabilities as POODLE (43%) and FREAK (20%). Heartbleed or POODLE over TLS or FREAK defines F valuation, POODLE over SSL3 defines C valuation[14]. Considering the SSL Server Rating the results show that the 35% of websites examined has F valuation, the 20% has C valuation, the 31% has B valuation and only 14% has A valuation.

The alarming results are that no website of public institutions has been valued as A and the 54% of them has a value of F. Also even more serious is the fact that the 25% of Italian bank websites has a valuation between C and F. The distribution of the valuations given in Table I for the three clusters considered (banks, institutions, others).The Table II shows the impact of Heartbleed, Poodle and Freak vulnerabilities on the three website clusters [11]. An important feature extracted is that 50% of the websites considered uses Apache engine and the 21% Microsoft IIS one.

TABLE I. SECURITY ASSESSMENT

	Metric			
	[A]	[B]	[C]	[F]
Banks	10	16	5	4
Institutions	0	6	10	19
Others	4	9	5	12
Total	14	31	20	35

During the test it has been possible to identify also other problems for these websites (2% of banks and 31% of public institutions). In fact some of them are characterized by the use of the SSL2, MIMT Insecure Renegotiation, OpenSSL CVE20140224 and Diffie-Hellman (DH) key exchange parameters.

 ISBN: 978-1-61208-427-5

TABLE II. VULNERABILITY ASSESSMENT

	Vulnerability		
	[HEARTBLEED]	[POODLE]	[FREAK]
Banks	0	6	2
Institutions	0	25	13
Others	1	13	5
Total	1	43	20

Another interesting test has been conducted by our team on PDA (point of access) for the Italian telematics process [8]. We have considered a dataset of 55 PDA clustered as Public Institutions, bar associations and Privates. The results shows that, despite the importance of the information exchanged through these channels, no website has obtained the A valuation, only 1 website (private) has obtained B valuation, 42 websites have obtained C valuation and 8 websites have obtained F valuation. For 4 websites it has not been possible to test the metric as not accessible. No website is affected by Heartbeat vulnerability, however, 89% of PDA is affected by POODLE vulnerability and 7% by FREAK. 100% of public PDA examined has a valuation between C and F (see the Table III below).

TABLE III. PDA SECURITY ASSESSMENT

	Metric			
	[A]	[B]	[C]	[F]
Bar Assoc.	0	0	39	0
Public	0	0	2	3
Private	0	1	1	5
Total	0	1	42	8

V. HEARTBLEED AND CLOUD RISKS

The vulnerability of the Heartbeat in OpenSSL can cause the following cloud risks and the severity level of these risks is very high.

A. Network Failure

The attacker can fail the internal CSPs switching and routing network may endanger connectivity for Cloud customer environments. Furthermore, the CSP may lose its control on external network connections. External connectivity is a critical part of the services offered. Data can be lost / damaged or network storage may be prohibited. Similar type of problem may occur if meta-information about data is lost.

B. Distributed Denial of Service (DDOS)

The Heartbleed vulnerability shall provide Cloud Server information which can be used to Distributed Denial of Service (DDOS) attacks on IP addresses within the network can easily harm the services of Cloud.

C. Loss of Customer Account and Configuration Data

Account settings and configuration data are essential in the process of service delivery. As mentioned above about the vulnerability, it can cause memory contents loss of customer accounts and configuration data can result in loss of service.

D. Data Interception

By exploiting Heartbleed vulnerability the attacker can intercept the data of the Cloud server machine. This situation will make Cloud Computing more vulnerable to attacks such as replay attacks, man-in-the-middle, spoofing, eavesdropping and sniffing.

E. Theft of Data

The attacker can theft the data of the Cloud users and he/she does not know what is going on behind the scene and they just suppose that they are transferring data to a secured Cloud Service Provider and their data is not intercepted.

F. Loss of Encryption Keys

The attacker could get private keys that sites use to encrypt and decrypt sensitive data. These keys are further used to encrypt all the traffic between Cloud customer and Cloud Service Provider. The attacker can get passwords of usernames and actual contents of data. This risk can further cause eavesdropping and theft of data.

G. Unauthorized Access

The active intruder after attacking on Cloud Service Provider website can gain unauthorized access to CSPs server machine and it may exploit the integrity and privacy of the customers. Fake users gain access to restricted areas.

H. Business Continuity

Due to Heartbleed attack, Cloud services can be blocked and Cloud customer may not become able to access it data over the Cloud network. The Cloud customer will also bear its financial loss as well as its business continuity will also be affected.

It is further added that during literature review, it is studied that Cloud Security Alliance in its report published on April 10, 2014 stated that after 24 hours of Heartbleed vulnerability discovery, 368 Cloud organizations were still vulnerable[10]. The Skyhigh recommended to the Cloud organizations to update OpenSSL and obtain new certificates. Furthermore, CSA has recommended five steps that every Cloud organization have to take in case infected Heartbleed.

VI. CONCLUSION AND FUTURE WORK

This paper has shown methods and tools to test the Heartbleed vulnerability. The results obtained by python code shows the importance and dangerousness of this vulnerability. Luckily, considering our dataset of 100 HTTPS public and private sector websites including Cloud websites and find out that only 1% of the websites is still affected by this vulnerability, meanwhile, POODLE and FREAK vulnerabilities are the new security problems to vanquish. An alarming scenario is that of the four banks characterized by POODLE, FREAK and MIMT vulnerabilities and of PDA (point of access) for Italian telematics process that shows as a better sensitivity about IT security problems is required. Interesting open topics to implement in future works are the extension of the analysis to other countries to compare the Italian results, the re-monitoring of these website farther on, a detailed analysis on POODLE and FREAK and a review on Shellshock attack.

 ISBN: 978-1-61208-427-5

REFERENCES

[1] Z. Kasten, D. Adrian, J. Halderman, and M. Bailey, 'The Matter of Heart-bleed', In Proceedings of the 2014 Conference on Internet Measurement Conference, ACM, pp. 475-488, 2014.

[2] M. Mshangi, 'Using Soft Systems Methodology and Activity Theory to Exploit Security of Web Applications against Heartbleed Vulnerability', International Journal of Computing & ICT Research, 8(2), pp. 32-52, 2015.

[3] J. A. Lewis, 'Heartbleed and the State of Cyber Security', Taylor and Frances Group, pp. 294-299, 2014

[4] E. Dreyfus, 'TLS hardening', BSD Magazine, 2014, [Retrieved on May, 2015].

[5] B. Chandra, 'A technical view of the OpenSSL Heartbleed vulnerability', IBM White Paper,2014, [Retrieved on May, 2015].

[6] US Department of Homeland Security, 'Heartbleed OpenSSL Vulnerability', National Cyber Security and Communications Integration Center, 2014, [Retrieved on May, 2015].

[7] Websense, 'OpenSSL Vulnerability CVS-2014-0160 (Heartbleed)', Web-sense White Paper, 2014, [Retrieved on May, 2015].

[8] V.Santarcangelo, G.Oddo, N.Santarcangelo, V.Ribaudo, and G.Lamacchia, 'Fattura elettronica, conservazione sostitutiva, processo telematico: Stru-menti per il miglioramento della qualita nella pubblica amminis-trazione', Sei Sigma e Qualita, RCE Multimedia vol.5, n.4 (2014).

[9] Qualys SSL Labs, 'SSL Server Rating Guide', 2014, [Retrieved on May, 2015].

[10] H. Byun, '24 Hours After Heartbleed, 368 Cloud Providers Still Vulnerable', CSA Security Alliance, 2014.

[11] B. Mller, T. Duong, and K. Kotowicz, 'This POODLE Bites: Exploiting the SSL 3.0 Fallback', Google Security Advisory, 2014.

[12] T. Duong and J. Rizzo, 'Here Come The Ninjas', 2011, [Retrieved on May, 2015].

[13] M.I. Tariq, 'Towards Information Security Metrics Framework for Cloud Computing', IJ-CLOSER, (2012).

[14] ISE, 'FREAK Security Advisory' , ISE CONFIDENTIAL, 2015, [Retrieved on May, 2015].

[15] Eelsivart, https://gist.github.com/eelsivart/10174134 , (2014), [Retrieved on May, 2015].

[16] M. R. Albrecht, D. Papini, K. G. Paterson, and R.Villanueva, 'Polanco Factoring 512-bit RSA Moduli for Fun (and a Profit of $9,000)', 2015, [Retrieved on May, 2015].

 ISBN: 978-1-61208-427-5