http://jates.org

Journal of Applied Technical and Educational Sciences jATES

ISSN 2560-5429

JATES Journal of Applied Technical and Educational Sciences

Some ethical hacking possibilities in Kali Linux environment

Petar Cisar[a], Robert Pinter[b]

[a]*University of Criminal Investigation and Police Studies, Cara Dusana 196, 11080 Zemun, Serbia, petar.cisar@kpu.edu.rs*

[b]*Subotica Tech - College of Applied Sciences, Marka Oreskovica 16, 24000 Subotica, Serbia, probi@vts.su.ac.rs*

Abstract

This paper deals with the problem of ethical hacking and security of computer systems. When we talk about security of an information system, we actually mean the primary three attributes of the system: confidentiality, integrity and availability. There are various approaches with aim to identify existing security weaknesses and security assessment. One of them is using Kali Linux operating system with its integrated effective tools specially adapted to the realization of various types of attacks. The paper gives a general overview of some Kali attacking possibilities on client and server side and highlights their specificities. The undoubted benefit of this operating system is a large collection of different hacking tools in one place which significantly facilitates vulnerability assessment and security testing.

Keywords: Kali Linux; tools; attack; security; ethical hacking

1. Introduction

In general, four main categories (or phases) of information security assessments can be identified (Hertzog, 2017): a vulnerability assessment, a compliance (audit) test, a traditional internal/external penetration test, and an application assessment. There are various methods with aim to identify existing security weaknesses and security assessment (Allen, 2014). One of them is using tools from Kali Linux operating system (OS).

Kali Linux is a Debian-based Linux distribution focused on advanced penetration testing and ethical hacking. It contains several hundred tools which are aimed at a wide range of information security tasks, such as penetration testing, security examinations, computer forensics and reverse engineering (Pritchett, 2013). The term hacking refers to identifying and exploiting security weaknesses in computer systems and/or networks.

Tools within Kali package are very diverse and can be divided into the following categories (Kali Linux Tools): Information gathering, Vulnerability analysis, Wireless attacks, Web applications, Exploitation tools, Forensics tools, Stress testing, Sniffing and spoofing, Password attacks, Maintaining attacks, Reverse engineering, Hardware hacking and Reporting tools (Fig. 1).

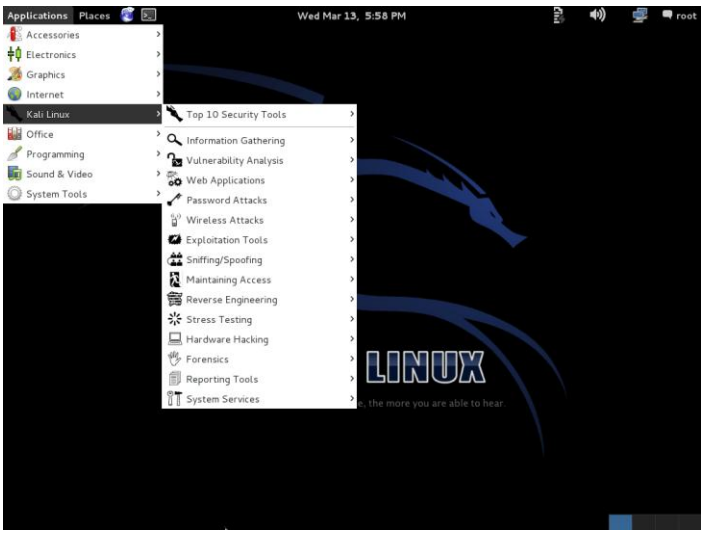

Fig. 1. Kali Linux integrated tools

Kali Linux contains frequently used security testing tools such as: Nmap (port scanner), Wireshark (packet analyzer), John The Ripper (password cracker), Aircrack-ng (software suite for penetration testing wireless LANs), Nikto (web server scanner), Sqlmap (tool for detecting and exploiting SQL injection flaws and taking over of database servers), Owasp-Zap (finding vulnerabilities in web applications), Metasploit Framework (exploitation) and many others.

In addition to Kali distribution as the most popular, other Linux distributions are also used for hacking (It's FOSS). They provide various tools that are needed for assessing networking security:

- BackBox is Ubuntu-based distribution developed for penetration testing and security assessment. It has own software repository providing latest stable versions of various system and network analysis toolkits and the best known ethical hacking tools. BackBox is designed with minimalism and uses XFCE (XForms Common Environment) desktop environment. It delivers a fast, effective and customizable work.

- Parrot Security OS is a relatively new hacking distribution. The target users are penetration testers who need cloud friendly environment with online anonymity and encrypted system. Parrot is also based on Debian and uses MATE as its desktop environment. A great number of tools for penetration testing are available here (along with some exclusive custom tools from Frozenbox Network).
- BlackArch is a penetration testing and security researching distribution built on Arch Linux. BlackArch has its own repository containing thousands of tools organized in various groups.
- Bugtraq is a distribution with a great range of penetration, forensic and laboratory tools. It is available with XFCE, GNOME and KDE desktop environments based on Ubuntu, Debian and OpenSUSE. Bugtraq contains a huge collection of penetration testing tools, mobile forensics and malware testing laboratories along with tools designed by the Bugtraq-community.
- DEFT (Digital Evidence & Forensics Toolkit) Linux is a distribution made for computer forensics, with the purpose of running live system without corrupting or tampering devices connected to the computer where the booting takes place. DEFT is combined with DART (Digital Advanced Response Toolkit), a forensics system for Windows OS. It uses LXDE desktop environment and WINE for running Windows tools.
- Samurai Web Testing Framework is developed with the sole purpose of penetration testing on web. Another difference from the previous distributions is that it comes as a virtual machine, supported by Virtualbox and VMWare. Samurai Web Testing Framework is based on Ubuntu and contains free and open source tools focusing on testing and attacking websites.
- Pentoo Linux is based on Gentoo Linux. It is a distribution focused on security and penetration testing and is available as Live CD with persistence support (any changes made in the Live environment will be available on the next boot if using a USB stick). Pentoo contains a number of customized tools and kernel features and uses XFCE desktop environment.
- CAINE (Computer Aided Investigative Environment) is completely focused on digital forensics. CAINE comes with a wide variety of tools developed for system forensics and analysis purpose.

- Network Security Toolkit is a bootable Live ISO (Live CD) based on Fedora. It provides a wide range of open source network security tools and has an advanced Web user interface for system / network administration, navigation, automation, network monitoring and analysis and configuration of many applications which can be found in this distribution.
- Fedora Security Spin represents a variation of Fedora designed for security auditing and testing and can also be used for teaching purpose. The main goal of this distribution is to help students and teachers in practicing and learning security methodologies on information security, web application security, forensics analysis etc.
- ArchStrike (former ArchAssault) is a distribution based on Arch Linux convenient for penetration testers and security professionals. It comes with all functionalities of Arch Linux, expanded with tools for penetration testing and cyber security. ArchStrike includes thousands of tools and applications, categorized into modular package groups.
- Other Linux hacking distributions: Cyborg Linux, Matriux, Weakerth4n etc.

Kali distribution was chosen for presentation in this paper because of its ease installation, ability to work in virtual environment, a large number of reliable security testing tools, and convenience for student training.

Attack is the basic form of hacking and can be defined as any action that compromises the security of information.

One of the most common vulnerability classes (attacks) are (Hertzog, 2017): denial-of-service (DoS; breaks the behavior of an application and makes it inaccessible), memory corruption (e.g. buffer overflow; leads to manipulation of process memory, often allowing an attacker code execution), Web vulnerabilities (which attack web services using techniques like SQL injection and XSS), password attacks (attacks against the authentication system; often leverage password lists to attack service credentials) and client-side attacks.

The process of network hacking can take many forms: pre connection attacks (packet sniffing, deauthentication attack), gaining access (cracking WEP/WPA/WPA2 encryption), post connection attacks (using network mapping with Nmap/Zenmap, Man-in-the-middle attacks, using of Wireshark, creating fake access points, spying, pivoting) and website hacking.

Speaking of ethical hacking, gaining access to computer device (personal computer, web server, network, mobile phone, TV and so on) is essential activity and can be practically realized by two different types of attack:

a) client side attack
b) server side attack

2. Client side attack

This type of attack requires some kind of user interaction, such as opening a specific file or a link. Information gathering is vital here, as well as creation and distribution of Trojans and use of social engineering to make target to run them. It is necessary to be positioned like a man-in-the-middle (MITM) - a network situation where the attacker is secretly placed between two participants, who believe they are directly communicating.

This type of attack is mostly launched in the following cases:

- If server side attacks fail (after unsuccessful attempts of using exploits in OS and application installed).
- If IP is probably useless (after pinging the target IP, the target stays hidden behind the router or a network).

Social engineering can be very useful for gathering information about the user(s) (Ex. name, Facebook account, password etc.), for building a strategy based on the information, to create backdoor based on information (the target runs the specific file or downloads some executables).

Protection against this type of attack (smart delivery methods) involves:

- Ensuring of not being in MITM situation - by usage of trusted networks or appropriate software (for instance, XArp).
- Only perform download from HTTPS (Hypertext Transfer Protocol Secure) pages.
- Checking file's MD5 signature (checksum) after download (for example, WinMD5Free tool makes it possible to compare original (provided by the developer or the download page) and current file's MD5 checksum values) - matching these values ensures that the file has not been modified or infected with backdoor malware.

One of the common forms of attack on the client side is the insertion of a Trojans into the client device. Existing of Trojans can be checked in many ways - manually or using a sandbox environment:

- Manually:
    a) Checking the properties of the suspicious file: the right click on file icon → Properties → Type of file. In this way, it can be determined whether the observed file is what it appears to be.

b) Resource Monitor - choice of Network option gives all the opened ports on the machine. Remote Address option displays all active IP addresses in that moment (Fig. 2). A suspicious (unknown) address should be identified among them. That address can be verified with Reverse DNS Lookup (lookup an IP address).

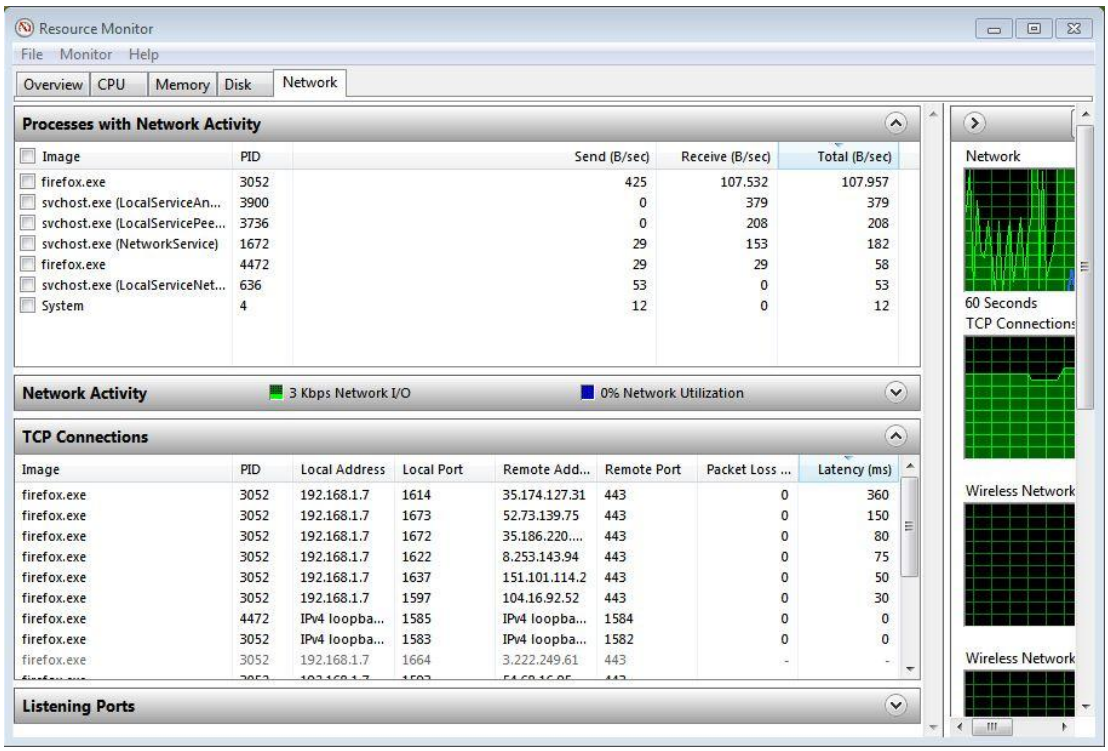

Fig. 2. Resource Monitor - identification of active TCP connections

- Running the file in a virtual machine and checking resources.
- Use of online sandbox service (malware analysis service) - a place where the file will be executed and analyzed with generating a detailed report (Fig. 3).

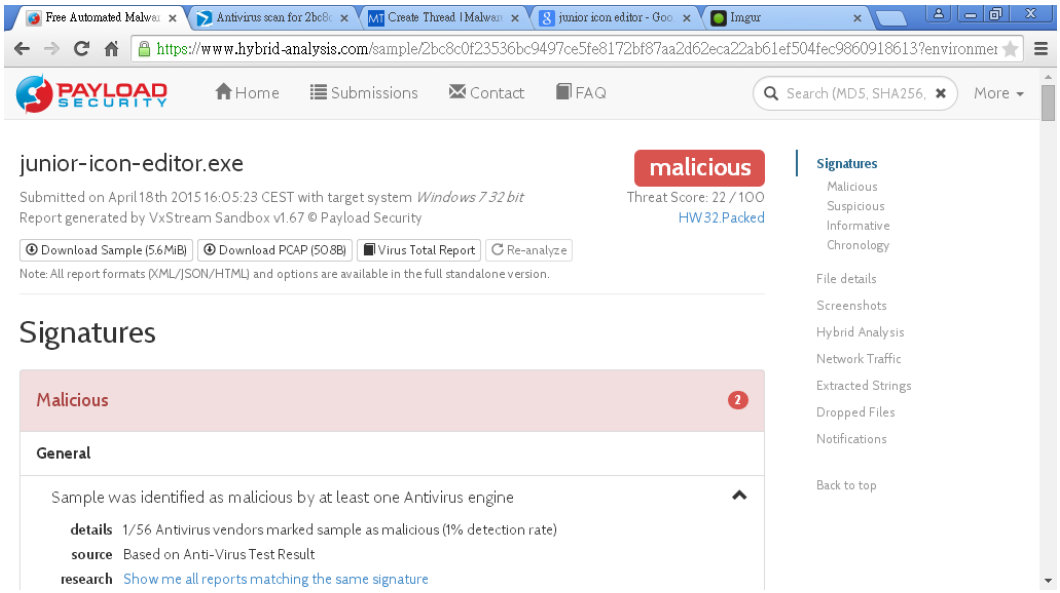

Fig. 3. Malware analysis (Hybrid Analysis)

3. **Server side attack**

This type of attack does not require any user interaction. All it takes is the target IP address. If this data is known, information gathering can start, followed by finding open ports, identification of operation systems, installed services and work from there.

Server side attack is very simple if identified target is on the same network (using tools like Netdiscover or Zenmap).

If a target has a domain, then running simple `ping` command will return its IP (for instance, `ping` www.facebook.com → 31.13.84.36).

Getting the IP is more complicated if the target is a personal computer. This might be useless if the target is accessing the internet through a network as the obtained IP will be the router's IP and not the target's. Client side attacks are more effective in this case as reverse connection can be used.

4. **Packet sniffing**

Packet sniffing is the activity of capturing packets of data flow across a computer network. The software or device used to do this is called a packet sniffer (Colasoft).

The process of packet sniffing in Kali Linux is a part of the Aircrack-ng suit (by `airodump-ng` sniffing tool). This tool is designed and used to capture all packets within range. It displays detailed information about networks (devices) around observed computer, connected clients etc. (Fig. 4).

Targeted packet sniffing is also supported and is based on BSSID (Basic Service Set Identifier) and channel or MAC address of the target.

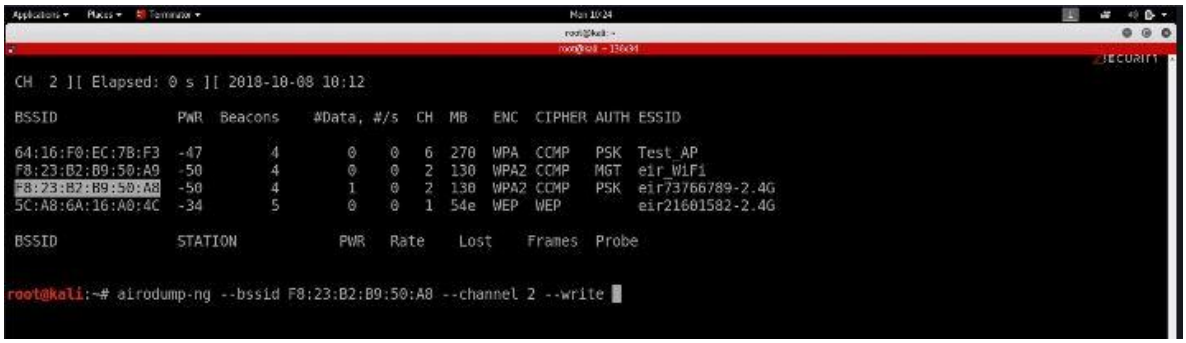

Fig. 4. Packet sniffing by *airodump-ng*

5. Deauthentication attack

Deauthentication attack is a type of attack which is focused on disconnecting any client (device) from any network (router). It belongs to the DoS (Denial-of-Service) attack category.

The main features of this type of attack are:

- Works on encrypted networks (WEP, WPA and WPA2).
- No need to know the network key.
- No need to connect to the network.

The attacker sends deauthenication packets (protocol - spoofed deauthentication message) to an access point, forcing the device to disconnect - telling it that it has been disconnected.

Example: `aireplay-ng --deauth 4(0 ili 1)`(number of authentication packets) `-a 00:10:18:90:2D:EE`(BSSID) `-c C0:18:85:C1:CF:01`(STATION) `mon0`

6. MITM attack - ARP poisoning theory

MITM attack is a general term for attack situation where an executor places him in a connection between a user and a web application - either to eavesdrop or to represent one of the parties, making it appear (establishing new connection) as if a normal information exchange is on-going (Imperva).

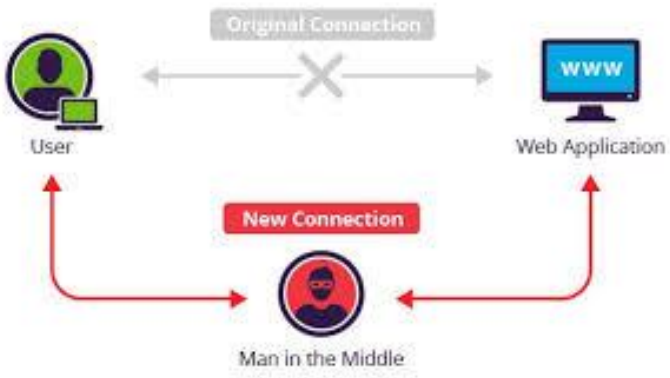

Fig. 5. Man-in-the-middle attack - basic principle

ARP spoofing is a type of attack in which a hacker sends falsified ARP (Address Resolution Protocol) messages over a local area network. This results in the linking of a hacker's MAC address with the IP address of a legitimate user or server on the network. Once the hacker's MAC address is connected to an authentic IP address, the hacker will begin receiving any data that is intended for that IP address. ARP spoofing can enable malicious persons to intercept,

modify or even stop data in-transit. ARP spoofing attacks can only occur on local area networks that utilize the ARP (Veracode).

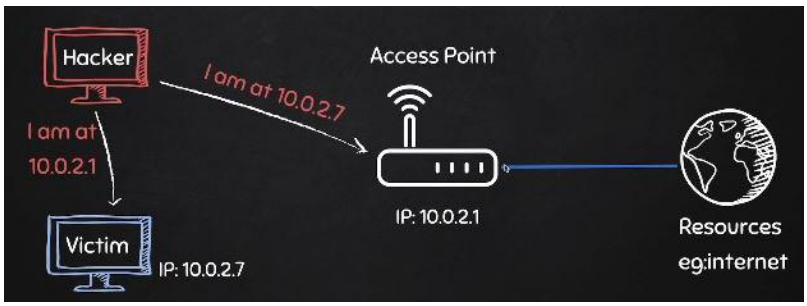

Fig. 6. ARP spoofing - example (Udemy)

ARP spoofing the following facts make possible:

1. Clients accept responses even if they did not send a request.
2. Clients trust response without any form of verification.

Prevention - Several methods can be used to prevent ARP poisoning, each with its own positives and negatives. These include static ARP entries (recommended for smaller networks), encryption (HTTPS, SSH), VPNs (VPN encrypt all of the data that travels between the client and the exit server), packet filters (packets that come from outside the network but contain source IP of inside the network should not be allowed) and software for detection of ARP Spoofing (for example, XArp). The most common detection criterion is unknown MAC address and host (marked in the figure below).

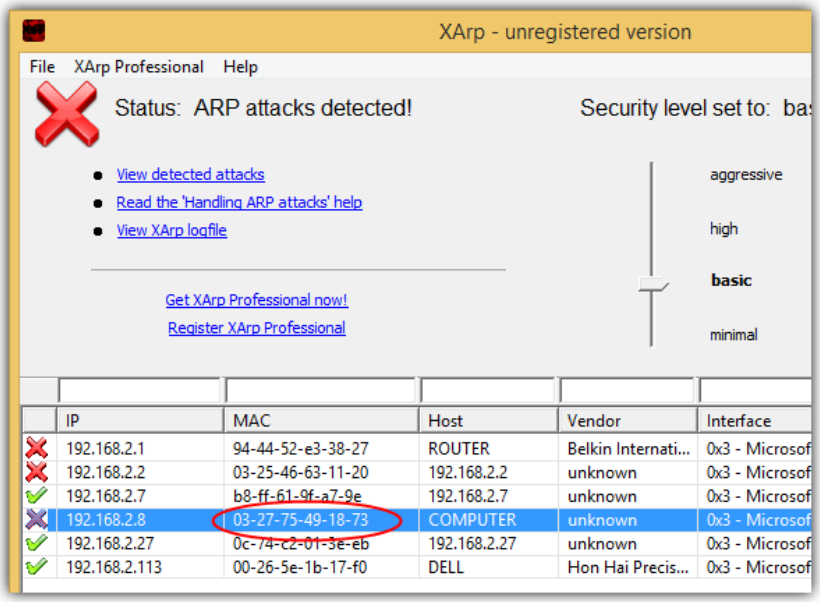

Fig. 7. XArp - ARP attack detection

7. MITM - Bypassing HTTPS

A general problem with HTTP protocol is that data is sent as plain text (the attacker is able to see usernames, passwords and all other sensitive data). This practically means that a MITM can read and edit requests and responses, causing unsecure communication.

Solutions for ensuring satisfactory security at the transport level:

- Using of HTTPS (HTTPS is an adaptation of HTTP).
- Encryption of HTTP using TLS (Transport Layer Security) or SSL (Secure Sockets Layer).

Problem that occurs with bypassing HTTPS is that most websites use HTTPS → sniffed data will be encrypted. Solution for this is to downgrade HTTPS to HTTP - by adequate using of `bettercap` program (network tool in Kali Linux for network capture, analysis and MITM attacks) and recorded caplets in HTTPS.

8. MITM - Bypassing HSTS

HTTP Strict Transport Security (HSTS) is a kind of web server security mechanism which over header informs user agents and web browsers that they should never load a site using HTTP and should automatically convert all attempts to access the site using HTTP to HTTPS requests instead (MDN Web Docs). HSTS is used by Facebook, Twitter and few famous websites.

A problem with bypassing HSTS is that modern browsers are hard-coded to only load a list of HSTS websites over HTTPS. Attempt to resolve this situation is to trick the browser into loading a different websites - replacing links for HSTS websites in HSTS caplets (.cap files) with similar (slightly modified) links (Ex. facebook.com → facebook.corn, twitter.com → twiter.com). Caplet is a configuration file containing a list of scripts - commands for interactive sessions. Running this file in Bettercap program will activate entered modifications (`hstshijack/hstshijack`).

Example: `hstshijack.cap`

```
set hstshijack.log
      /usr/share/bettercap/caplets/hstshijack/ssl.log

set hstshijack.ignore   *

set hstshijack.targets
      twitter.com,*.twitter.com,facebook.com,*.facebook.com
```

```
set hstshijack.replacements
      twitter.corn,*.twitter.corn,facebook.corn,*.facebook.corn

set hstshijack.obfuscate      false

set hstshijack.encode         false

set hstshijack.payloads
      *:/usr/share/bettercap/caplets/hstshijack/payloads/keylogger.js


set http.proxy.script         /usr/share/bettercap/caplets/hstshijack/
hstshijack.js

set dns.spoof.domains
      twitter.corn,*.twitter.corn,facebook.corn,*.facebook.corn

http.proxy  on

dns.spoof   on
```

9. **MITM - DNS spoofing attack**

DNS cache poisoning (also known as DNS spoofing) is a type of attack that exploits vulnerabilities in the domain name system (DNS) to divert traffic away from legitimate website and towards fake ones (Fig. 8). The attack principle is based on falsifying DNS records with aim of traffic redirection. One of the reasons DNS poisoning is dangerous is because it can spread from DNS server to DNS server.

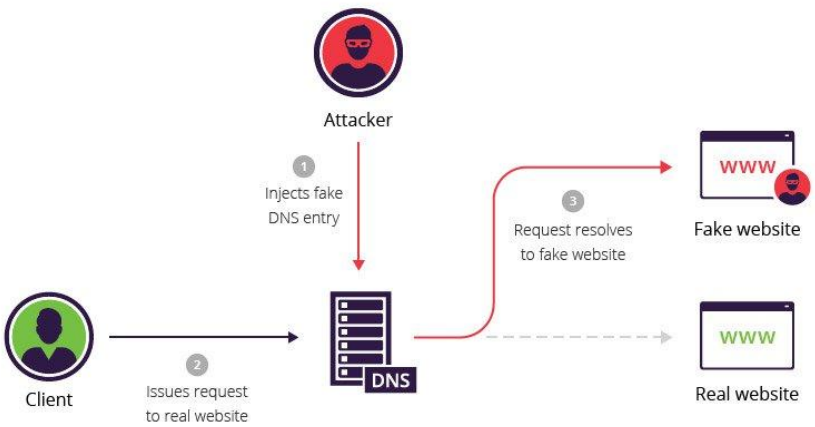

Fig. 8. DNS spoofing attack (Imperva)

Various tools can be used to launch this attack. Arpspoof from Kali Linux collection is one of them. The attacking procedure using this program consists of the following steps:

1. Finding own default gateway - `#ip route`
2. Finding the network interface - `#ifconfig`
3. Finding the IP address of victim - `#ifconfig or netdiscover -r Default Gateway`
4. Starting the ARP poisoning/spoofing - `#arpspoof -i [Network Interface Name] -t [victim IP] [Router IP]/[-r Default Gateway]`

where `i` is for interface, `t` is for target and `r` is for default gateway.

During ARP spoofing the target has no internet connection. When the attack is stopped, the internet connection starts working again.

10.**MITM - code injection attack**

Code injection is the activity that enables the attacker to execute some specific code as a consequence of security vulnerabilities in web applications. Attacking possibilities depend on the limitations of the server-side interpreter (Python, Ruby, ASP, PHP, etc.). There are a few types of code injection attacks (The Security Buddy): SQL injection, HTML (JavaScript) injection, Dynamic code evaluation, File inclusion, Shell injection or Command injection.

One of the common forms of this attack is JavaScript code injection (can be realized by Bettercap program) in loaded pages. Code gets executed by target browser - the situation called remote code execution (RCE).

Code injection can be used to:

- ✓ replace links
- ✓ replace images
- ✓ insert HTML elements
- ✓ hook target browsers to exploitation frameworks
- ✓ ...

11.**Creating a fake access point (honeypot)**

A fake access point (AP), also known as honeypot, is an access point which broadcasts its signal the same way a router and even works like a router does but in reality it gathers packets from its clients which effectively means all data is streamed through the honeypot and the packets are open to modification and sniffing (Medium).

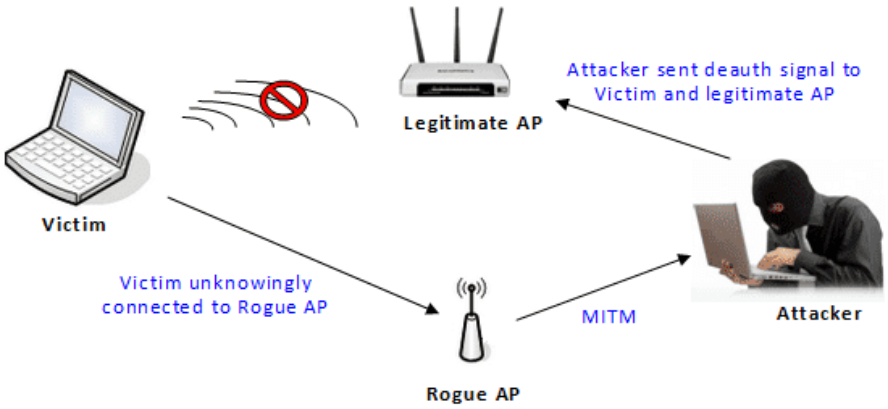

Fig. 9. Fake access point (Medium)

Mana-toolkit is a set of tools that run rogue access point attacks and wireless MITM. It can:

- ✓ Automatically configure and create fake AP.
- ✓ Automatically sniff data.
- ✓ Automatically bypass HTTPS.
- ✓ etc.

Mana has three main start scripts:

`start-noupstream` - starts an AP with no internet connection

`start-nat-simple` - starts a regular AP using internet connection in the upstream interface

`start-nat-full` - starts AP with internet connection and also starts sslstrip, sslsplit, firelamp and attempts to bypass HSTS.

12.**MAC address and the ability of its modification**

A MAC (Media Access Control) address is a hardware identification number that uniquely identifies each device on a network. The MAC address is manufactured into every network card (Ethernet card or Wi-Fi card), and therefore cannot be changed (Tech Terms).

MAC address is:

- Permanent
- Physical
- Unique

In Kali Linux, an easy way to determine the MAC addresses of installed network cards is to execute a command `ifconfig` for network interface configuration (Fig. 10).

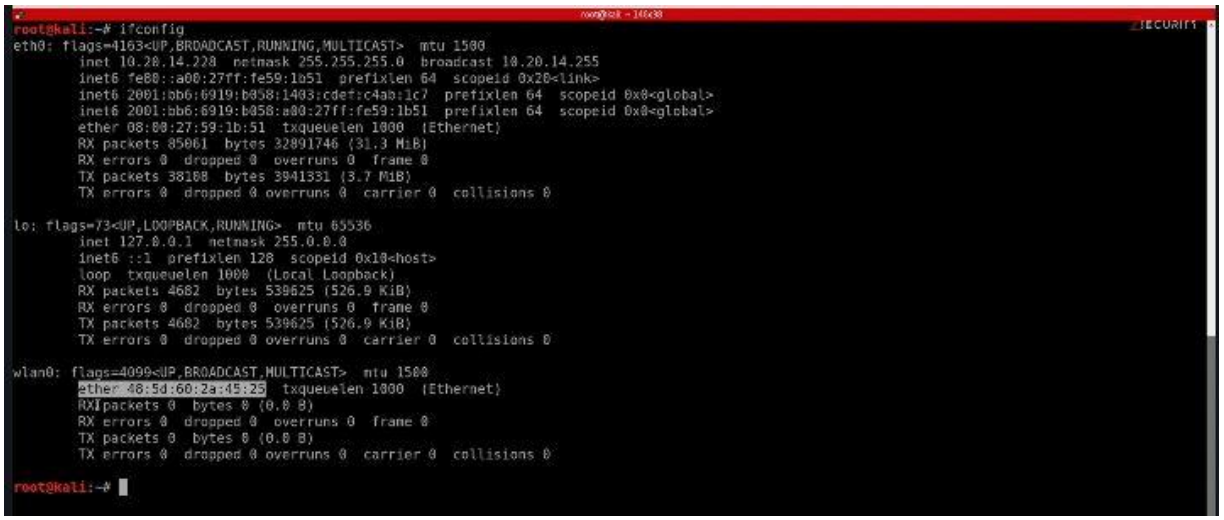

Fig. 10. MAC address determination (`ifconfig`)

In certain hacking situations, it is necessary to temporarily change the MAC address (in memory). The reasons for this could be:

- increase anonymity
- impersonate other devices
- bypass filters

The change process consists of the following steps:

1. Disable the interface (`ifconfig wlan0 down`).

2. Change the option (`ifconfig wlan0 hw ether 00:11:22:33:44:55; ifconfig wlan0 up`)

Restarting (reset) the computer brings back the original (physical) MAC address.

13.**Post exploitation (after gaining access)**

One of the most common post exploitation activities are:

- spying - capturing key strikes and taking screenshots of the target computer.
- pivoting - using a hacked device as a pivot, with aim to gain access to other devices in a network by Autoroute program (for setting up a route between hacker and hacked device, which gives hacker access to devices on the network.).

After exploiting a system there are two different approaches that can be applied - either smash and grab or low and slow. One tool which can be used for low and slow information gathering is the keystroke logger script with Meterpreter. This tool is very well designed, allowing capturing all keyboard inputs from the system, without writing anything to disk, leaving a minimal forensic footprint for investigators to later follow up on. It is perfect for getting

passwords, user accounts, and all sorts of other valuable information (https://www.offensive-security.com/metasploit-unleashed/keylogging/).

For instance:

`>keyscan_start` - shows current working directory

`>keyscan_dump` - lists files in the current working directory

`>keyscan_stop` - changes working directory to [location]

`>screenshot`

14. **Website hacking**

A website can be hacked on different ways, depending on what the hacking activities are oriented to:

- Attack on application installed on a computer → web application pentesting
- Attack on computer that uses an OS + other applications → server side attacks
- Attack on Website managed by humans (administrators) → client side attacks

Website hacking consists of several phases:

1. Information gathering (IP address, domain name information, used technologies, other websites on the same server, DNS records, unlisted files, subdomains, directories)

Important activities from this phase are: gathering basic information, discovering technologies used on the website, gathering comprehensive DNS information, discovering websites on the same server, discovering subdomains, and discovering and analyzing discovered sensitive files.

Useful tools used for this purpose:

- Whois Lookup - basic information about the owner of the target (Whois Lookup)
- Netcraft Site Report - shows technologies used on the target (Netcraft)
- Robtex DNS Lookup - shows comprehensive information about the target website (Robtex)
- `dirb` (Web content scanner) - `dirb [target][wordlist][options]`

2. File upload, code execution and file inclusion vulnerabilities

Important activities from this phase are:

- Discovering and exploiting file upload vulnerabilities - allow users to upload executable files (such as php) - tool `weevly` (generate backdoor, upload generated file and connect to it)

- Discovering and exploiting code execution vulnerabilities - allows an attacker to execute OS commands, Windows or Linux commands, can be used to get a reverse shell or upload any file using `wget` command, code execution commands attached in the resources.

- Discovering and exploiting local file inclusion vulnerabilities

- Discovery and exploitation of remote file inclusion vulnerabilities

Prevention from these vulnerabilities:

- File upload vulnerabilities - Only allow safe files to be uploaded - not php or any executables
- Code execution vulnerabilities - don't use dangerous functions (that use OS, filter user input before execution)
- File inclusion - disable `allow_url_fopen` and `allow_url_include`

3. SQL injection vulnerabilities

Most websites use a database to store data. Most data stored in it have sensitive character (usernames, passwords, pictures etc.). Web application reads, updates and inserts data in the database. Interaction with database is done by the language called SQL (Structured Query Language).

Important activities from this phase are:

- Discovering SQL injections in POST/GET

- Bypassing logins using SQL injection vulnerability - for example, `username='admin' and password='aaa' or 1=1 #'`

- Finding and reading database tables

- Extracting sensitive data - passwords

- Reading and writing files on the server using SQL injection vulnerability

- Discovering SQL injections and extracting data using SQLmap

Prevention from these vulnerabilities: using of parameterized statements in (server side language) code, separate data from SQL code.

4. Cross-site scripting (XSS) vulnerabilities

XSS allow an attacker to inject JavaScript code into the page. This code is executed on the client machine (not the server) when the page loads. There are three main types of XSS: persistent / stored (the injected code is stored in database), reflected (the code is only executed when the target user runs specific URL written and sent by attacker) and DOM based (results from JavaScript code written on the client machine).

Important activities from this phase are: discovering reflected and stored XSS and exploiting XSS - hooking vulnerable page visitors to BeEF (Browser Exploitation Framework - a penetration testing tool that focuses on the web browser).

Prevention from these vulnerabilities includes minimizing the usage of untrusted user input on HTML and escaping any untrusted input before inserting it into the page.

5. Discovering vulnerabilities automatically - OWASP ZAP (Open Web Application Security Project - Zed Attack Proxy)

This is a tool for scanning target website for vulnerabilities and analyzing scan results - the target URL needs to be entered (Fig. 11).

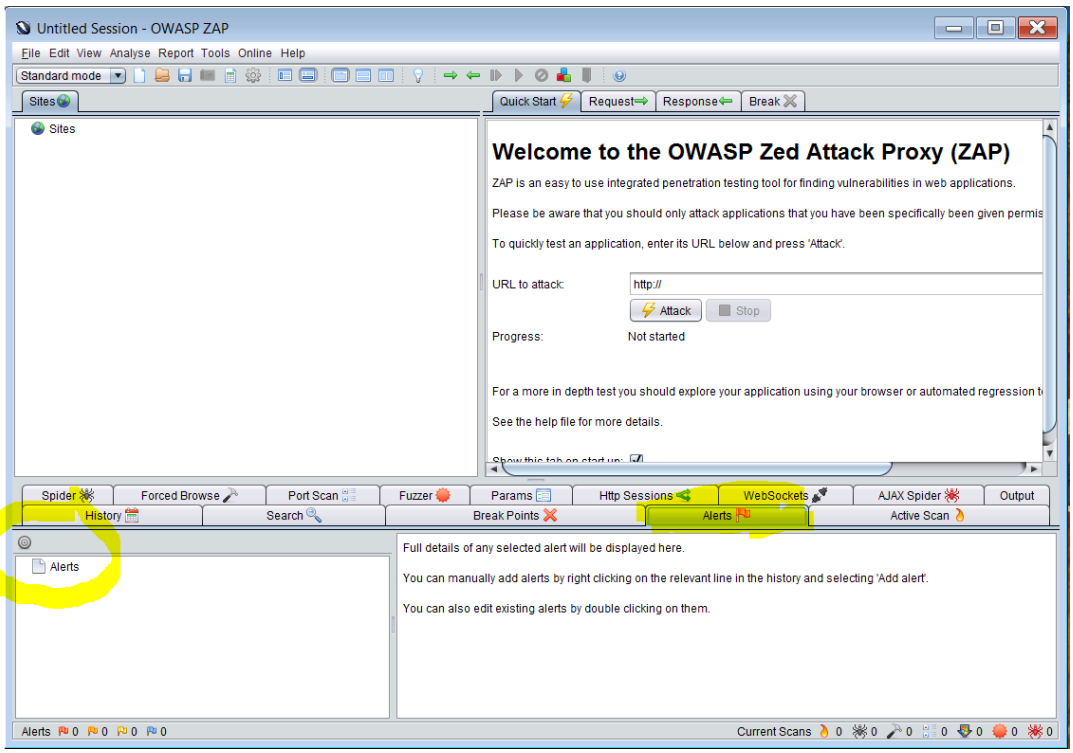

Fig. 11. ZAP (main screen)

For instance, the web application penetration testing methodology based on OWASP consists of 12 subcategories (OWASP): 1. Introduction and Objectives 2. Information Gathering 3. Configuration and Deploy Management Testing 4. Identity Management Testing 5. Authentication Testing 6. Authorization Testing 7. Session Management Testing 8. Data Validation Testing 9. Error Handling 10. Cryptography 11. Business Logic Testing 12. Client Side Testing.

15.**Best Kali tools**

The best Kali tools can be summarized in the following table (Linuxhint).

Table 1. The best Kali tools

Name	Function
Metasploit Framework	modules for automation the process of exploiting
Wireshark, Bettercap	sniffing and spoofing
Social Engineering Toolkit (SET)	exploitation tool
Aircrack-NG Suite	wireless attack
THC Hydra	online password cracker
John The Ripper	offline password cracker
Crunch	utility to create custom wordlists
Hash-Identifier and findmyhash	password attacks
SQLMap	detecting and exploiting SQL injection vulnerabilities
JoomScan & WPScan	tool to scan and analyze Joomla / WordPress CMS (content management system)
Httrack	website / webpage cloner
OWASP-ZAP	testing web application security
Burp Suite	mapping and analysis of an application's attack surface, finding and exploiting security vulnerabilities
SQLiv	SQL injection scanner
Nikto	vulnerability analysis
Dirbuster (Dirb)	tool to find hidden objects, files and directories on a website
NMap	network discovery and

	security auditing
Maltegoce (Maltego Community Edition)	tool to discover and collect data about the target and visualizes that collected data into graph for analysis
Whois	querying databases that store the registered users of an Internet resource (domain name or an IP address block)
WhatWeb	website identification, including CMS, blogging platforms, statistic/analytic packages, JavaScript libraries, web servers, and embedded devices
Traceroute	displaying the connection route and measuring transit delays of packets across an IP network
Proxychains	cover and handle whatever job
MacChanger	changing the MAC address

16. **Conclusions**

Kali Linux is an OS with numerous integrated effective tools specially adapted to the realization of various types of attacks. The paper emphasized the importance of Kali attacking possibilities in form of pre connection attacks, gaining access, post connection attacks and website hacking and highlighted their specificities. The undoubted benefit of this OS is a large collection of different hacking tools in one place which significantly facilitates vulnerability assessment and security testing.

This OS is open source system and can be easily accessed by the users. All the codes in the Kali Linux can be viewed easily by anyone and the open development Git tree makes easy to view the development of coding at every step.

Kali adheres to the FHS (File-system Hierarchy Standard), allowing Linux users to easily locate binaries, support files, libraries, etc. This is the very important feature of the Kali Linux that makes it stand out among the other Linux systems.

References

Allen, L., Heriyanto, T., Ali, S. (2014). *Kali Linux - Assuring Security by Penetration Testing*. Packt Publishing. 54-64.

Hertzog, R., O'Gorman, J., Aharoni, M. (2017). *Kali Linux Revealed*. Offsec Press. 283-284.

Pritchett, W., De Smet, D. (2013). Kali Linux Cookbook, Packt Publishing.

Colasoft https://www.colasoft.com/resources/packet_sniffing.php

Hybrid Analysis https://www.hybrid-analysis.com/

Imperva https://www.imperva.com/learn/application-security/man-in-the-middle-attack-mitm/

Imperva https://www.imperva.com/learn/application-security/dns-spoofing/

It's FOSS https://itsfoss.com/linux-hacking-penetration-testing/

Kali Linux Tools https://tools.kali.org/tools-listing

Kovari, A., & Dukan, P. (2012). KVM & OpenVZ virtualization based IaaS open source cloud virtualization platforms: OpenNode, Proxmox VE. IEEE 10th Jubilee International Symposium on Intelligent Systems and Informatics, pp. 335-339.

Linuxhint https://linuxhint.com/top-25-best-kali-linux-tools/

MDN Web Docs https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Strict-Transport-Security

Medium https://medium.com/@arnavtripathy98/how-to-make-a-fake-access-point-with-mana-toolkit-2464c1843d1e

Netcraft https://toolbar.netcraft.com/site_report

Offensive Security: Penetration Testing With Kali Linux https://www.offensive-security.com/documentation/penetration-testing-with-kali.pdf

Official Kali Linux Documentation https://docs.kali.org/pdf/kali-book-en.pdf

OWASP Web Application Penetration Testing
https://www.owasp.org/index.php/Web_Application_Penetration_Testing

Robtex DNS Lookup https://www.robtex.com/

Tech Terms https://techterms.com/definition/macaddress

The Security Buddy https://www.thesecuritybuddy.com/vulnerabilities/what-is-code-injection-attack/

Udemy https://www.udemy.com/learn-ethical-hacking-from-scratch/

Veracode https://www.veracode.com/security/arp-spoofing

Whois Lookup http://whois.domaintools.com/

About Authors

Petar Cisar earned a PhD degree in Information Systems. He works at the University of Criminalistic and Police Studies in Belgrade as associate professor. The spheres of his interest are information technology, computer and telecommunications networks, network security,

soft computing and computation intelligence. He is IEEE member - Computer Society Technical Committee on Security and Privacy and a member of the International Society for the implementation of fuzzy-theory with its seat in Budapest, as well as an external member of the public body of Hungarian Academy of Sciences and Arts. In addition, he is a member of Serbian Chamber of Engineers.

Robert Pinter is a professor at Subotica Tech - College of Applied Sciences. He obtained his MSc degree at the Electrical Engineering Faculty at the Budapest University of Technology. He defended his PhD thesis at the Technical Faculty "Mihajlo Pupin" in Zrenjanin, Serbia, in 2012. He teaches various computer science courses in the field of programming languages and mobile application development. The main research area in his scientific works is improving the efficiency of e-learning with the application of novel technologies and new teaching methodology.