***EXPERIMENT NO:01***

***Aim:*** Case study on NetStumbler and Kismet.

***Theory:***

NetStumbler -

NetStumbler is a tool for Windows that facilitates detection of Wireless LANs using the 802.11b, 802.11a and 802.11g WLAN standards. It runs on Microsoft Windows operating systems from Windows 2000 to Windows XP. A trimmed-down version called Ministumbler is available for the handheld Windows CE operating system.

The program is commonly used for:

• War driving

• Verifying network configurations

• Finding locations with poor coverage in a WLAN

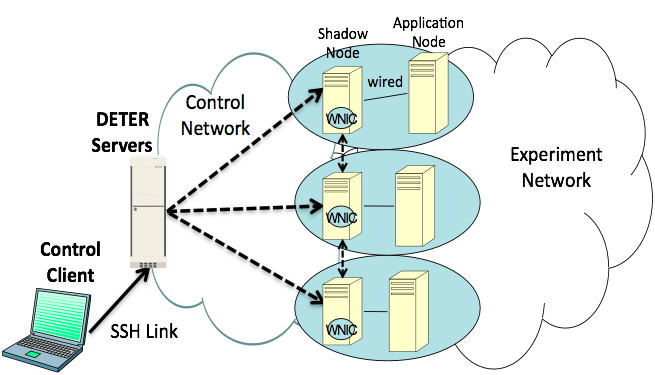
• Detecting causes of wireless interference

• Detecting unauthorized ("rogue") access points

• Aiming directional antennas for long-haul WLAN links

No updated version has been developed since 2005.

Architecture of NetStumbler:



key features of “NetStumbler”:

· Verify that your network is set up the way you intended.

· Find locations with poor coverage in your WLAN.

· Detect other networks that might be causing interference with your network.

· Detect unauthorized “rogue” access points in your workplace.

· Help aim directional antennas for long-haul WLAN links.

· Use it recreationally for WarDriving.

Details about the WLANs detected are displayed in several columns as explained below.

**SSID** – the service set identifier or name of the WLAN.

**Channel** – the channel that the WLAN is operating on.

**RSSI** – the received signal strength indicator, or measurement of the power (signal strength) present in a received radio signal. The lower, the stronger (for example, my WLAN whose AP is ten feet away is rated -36 while more distant WLANs are -67 to -88).

**Security** – the type of encryption the WLAN uses to secure its transmissions. ‘None’ means no encryption is used leaving that WLAN wide open. Other possibilities are WEP, WPA-TKIP (aka WPA Personal), and RSNA -CCMP which is the same as WPA2-CCMP.

**MAC** **Address** – the MAC address of the wireless access point (AP).

**Max** **Rate** – the theoretical maximum data transmission rate for the WLAN. 802.11g can theoretically reach 54 megabits per second (Mbps); 802.11n can surpass even this speed.

**Vendor** – the manufacturer of the wireless AP, such as Linksys, Netgear, D-Link, etc.

**Network** **Type** – only two choices here: infrastructure or ad hoc. In infrastructure mode, wireless-capable devices communicate through an access point that serves as a bridge to a wired network infrastructure. Ad hoc mode enables direct peer-to-peer transmissions between wireless-capable devices without the intervention of an AP.

**First** **Seen** – displays the time when inSSIDer first detected the WLAN.

**Last** **Seen** – displays the time when inSSIDer last saw the WLAN (presumably before it went offline).

**Latitude** **and** **Longitude** – used in conjunction with inSIDer's GPS functionality (described below).

Requirements

General Requirements:

The requirements for NetStumbler are somewhat complex and depend on hardware, firmware versions, driver versions and operating system. The best way to see if it works on your system is to try it. Some configurations have been extensively tested and are known to work. These are detailed at http://www.stumbler.net/compat. If your configuration works but is not listed, or is listed but does not work, please follow the instructions on the web site. The following are rules of thumb that you can follow in case you cannot reach the web site for some reason. This version of NetStumbler requires Windows 2000, Windows XP, or better. The Proxim models 8410-WD and 8420-WD are known to work. The 8410-WD has also been sold as the Dell Truemobile 1150, Compaq WL110, Avaya Wireless 802.11b PC Card, and others. Most cards based on the Intersil Prism/Prism2 chipset also work. Most 802.11b, 802.11a and 802.11g wireless LAN adapters should work on Windows XP. Some may work on Windows 2000 too. Many of them report inaccurate Signal strength, and if using the "NDIS 5.1" card access method then Noise level will not be reported. This includes cards based on Atheros, Atmel, Broadcom, Cisco and Centrino chip sets.

Firmware Requirements:

If you have an old Wave LAN/IEEE card then please note that the Wave LAN firmware (version 4.X and below) does not work with NetStumbler. If your card has this version, you are advised to upgrade to the latest version available from Proxim's web site. This will also ensure compatibility with the 802.11b standard.

Other Requirements and Compatibility Issues:

Your card must be configured in such a way that it can be seen by the management software that came with the card. The Microsoft-provided Orinoco drivers that come with Windows 2000 do not work with NetStumbler. Please visit Windows Update or www.proxim.com and upgrade to the latest drivers. When NetStumbler is in "auto reconfigure" mode (the default), it will occasionally disconnect you from your network. This enables it to perform its scans accurately, and is not a bug. If you have the WLAN card configured to connect to a specific SSID, NetStumbler may not report any access points other than those that have that SSID. Configure your card with a blank SSID or, if a blank one is not permitted, "ANY" (without quotes).

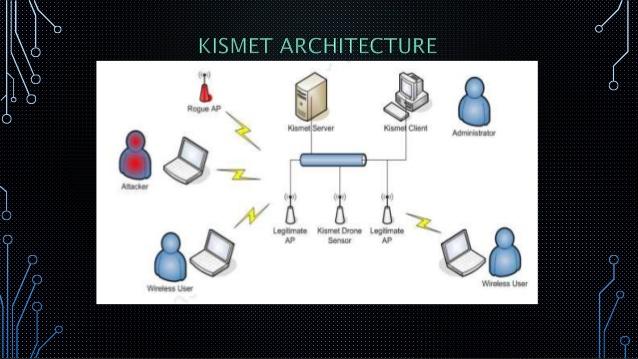
Kismet:

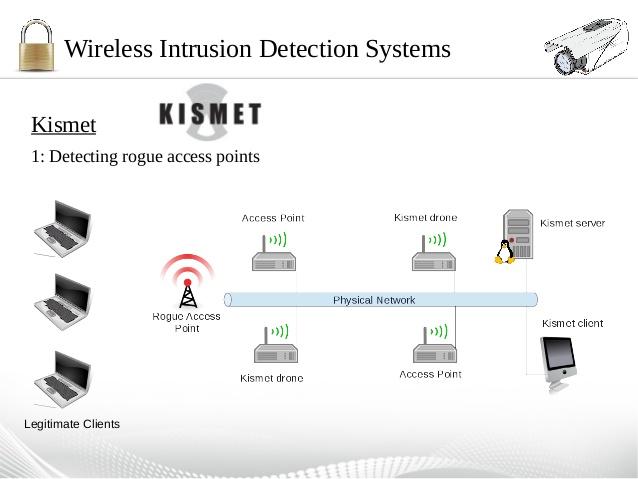
**Kismet** is a network detector, packet sniffer, and intrusion detection system for 802.11 wireless LANs. Kismet will work with any wireless card which supports raw monitoring mode, and can sniff 802.11a, 802.11b, 802.11g, and 802.11n traffic. The program runs under Linux, Free BSD, Net BSD , Open BSD, and Mac OS X. The client can also run on Microsoft Windows, although, aside from external drones there's only one supported wireless hardware available as packet source.

Kismet is an 802.11 wireless network detector, sniffer, and intrusion detection system. Kismet will work with any wireless card which supports raw monitoring mode, and can sniff 802.11b, 802.11a, 802.11g, and 802.11n traffic (devices and drivers permitting).

Kismet also sports a plugin architecture allowing for additional non-802.11 protocols to be decoded. Kismet identifies networks by passively collecting packets and detecting networks, which allows it to detect (and given time, expose the names of) hidden networks and the presence of non-beaconing networks via data traffic.

**Architecture of Kismet:**

****

****

Features:

Kismet differs from other wireless network detectors in working passively. Namely, without sending any loggable packets, it is able to detect the presence of both [wireless access points](https://en.wikipedia.org/wiki/Wireless_access_point) and wireless clients, and to associate them with each other. It is also the most widely used and up to date open source wireless monitoring tool. Kismet also includes basic wireless IDS features such as detecting active wireless sniffing programs including NetStumbler, as well as a number of wireless network attacks. Kismet features the ability to log all sniffed packets and save them in a tcpdump/ Wireshark or Airsnort compatible file format. Kismet can also capture "Per-Packet Information" headers. Kismet also features the ability to detect default or "not configured" networks, probe requests, and determine what level of wireless encryption is used on a given access point. In order to find as many networks as possible, Kismet supports channel hopping. This means that it constantly changes from channel to channel non-sequentially, in a user-defined sequence with a default value that leaves big holes between channels (for example, 1-6-11-2-7-12-3-8-13-4-9-14-5-10). The advantage with this method is that it will capture more packets because adjacent channels overlap.

Kismet also supports logging of the geographical coordinates of the network if the input from a GPS receiver is additionally available.

## Server / Drone / Client infrastructure:

Kismet has three separate parts. A *drone* can be used to collect packets, and then pass them on to a *server* for interpretation. A server can either be used in conjunction with a drone, or on its own, interpreting packet data, and extrapolating wireless information, and organizing it. The *client* communicates with the server and displays the information the server collects.

## Plugins:

With the updating of Kismet to -ng, Kismet now supports a wide variety of scanning plugins including DECT, Bluetooth, and others. Kismet is an 802.11 wireless network detector, sniffer, and intrusion detection system. Kismet will work with any wireless card which supports raw monitoring mode, and can sniff 802.11b, 802.11a, 802.11g, and 802.11n traffic (devices and drivers permitting).

Kismet also sports a plugin architecture allowing for additional

non-802.11 protocols to be decoded. Kismet identifies networks by passively collecting packets and detecting networks, which allows it to detect (and given time, expose the names of) hidden networks and the presence of non-beaconing networks via data traffic.

## Usage:

Kismet is used in a number of commercial and open source projects. It is distributed with Kali Linux. It is used for wireless reconnaissance,and can be used with other packages for an inexpensive wireless intrusion detection system. It has been used in a number of peer reviewed studies such as "Detecting Rogue Access Points using Kismet" .Kismet is a utility that can be placed on the network passively, meaning that a security team can look at data immediately, should the need arise. Another great feature of Kismet is that it can connect via Bluetooth to a computer or smartphone with a GPS, and show the location of each detected network. This is especially useful on campuses where there might be unauthorized wireless networks, because security teams can see exactly where the network comes from. With Kismet, incident response teams get a free, lightweight tool that can help identify potentially harmful access points in a more effective way.

***Conclusion:*** Hence we have studied case study of netstumbler and kismet.