Wireless Honeypot: Framework, Architectures and Tools

Radhika Goel, Anjali Sardana, and R. C. Joshi
(Corresponding author: Radhika Goel)

Department of Electronics and Computer Science Engineering, IIT Roorkee, India
(Email: radhikaiitr@gmail.com)
(Received Aug. 23, 2010; revised and accepted Mar. 2, 2011)

Abstract

Even though a spectrum of security solutions exists, the lack of knowledge about the exploitation methods used to compromise wireless networks is threatening the free and easy usage of wireless technologies in commercial world. Wireless Honeypot has come up as a recent solution to evaluate and assess the security in wireless environment at different layers. This paper surveys a range of wireless honeypot systems and tools that have been proposed and deployed. Then, it presents a general framework for wireless honeypot systems that encompasses a broad range of honeypot architectures, and categories previous systems according to that framework, highlighting the results of those projects. The results show that though an array of wireless honeypot models exists, none of them is able to provide full protection in real-time environment. There is a need for worldwide deployment of honeypots incorporating all the three- deceptive, detection, and deterrence mechanisms, to control the wireless attack scenario.

Keywords: Deception, detection, honeypots, network security, survey, wireless

1 Introduction

Today, wireless devices are found everywhere from IEEE 802.11 standard based laptops to Bluetooth enabled mobile phones. The advancement in wireless technologies has made these wireless devices an indispensible part of our daily life. However, wireless networks are more susceptible to intrusion than their wired counterparts. A crack in wireless security jeopardizes not only the wireless network but also the wired network connected to it.

There are many open or non-secured APs everywhere. Many prevention and detection techniques have been developed for wireless security like MAC Address Filtering, Static IP Addressing, 802.11 security standards, WEP, WPA, RF Shielding etc. However, the current state of WLAN encryption and authentication is so weak that hackers have found ways to break through these securities [15, 20]. Existing technologies are prone to attacks like Client-Client Attacks, Jamming Attack, Rogue Access points, War Diving, Inception, Interception Strategies, Insertion Attack, Misconfiguration Attack, Incorporating SSID and so on [1, 21]. Moreover, the lack of necessity of any physical connection makes escape easy and safe for the attacker and, remove any fear from his conscience.

For maintaining higher standards of security, the need to keep an open eye on hacker's action all the time was felt. In 2002, Kevin Poulsen introduced the concept of honeypot to wireless domain. Honeypot technology is a technology that helps us to get information about hacker, his skill level, his frequency of attacks, his goals and methods used to access the security of current devices by deception. In wireless domain, a wireless resource deployed to lure the attacker is called wireless honeypot. In Lance Spitzner [18] terms "It's a wireless resource whose value lies is it's unauthorized or illicit use." This resource can be real devices/applications (High Interaction Honeypot) or can be devices that emulate the behavior of real device (Low Interaction Honeypot).

There are three defense mechanisms of Honeypots that are discussed in this paper - deception, detection and deterrence. In deception mechanism, systems are made to look as productive as possible but in reality they do not have any valuable information to give to attacker. Honeypots may be in the form of emulated virtual environments or real systems, but they don't have any real production work to serve. They just keep the attacker busy, utilize his resources and study his techniques thoroughly. These attack patterns are then used for incorporation into intrusion detection systems (IDS) as attacks rules. Honeypots also helps in detecting attackers. They are deployed with real production systems but they do not expect any legitimate activity. Therefore, any traffic on such systems is suspected as an attack which is then logged and analyzed. In deterrence technique, honeypots reveal their presence in the network but not their location. Such systems create fear in the attacker and deter him from committing any illegal activity. For the effectiveness of this technique, it is important that attacker should be able to just fingerprint the presence of deception and must not be able to

find the exact location of honeypot in the network.

In June 15^{th}, 2002, Science Applications International Corporation (SAIC) in Washington DC (US) launched the first organized wireless honeypot- WISE (Wireless Information Security Experiment) [13]. The focus of this initial research was to get statistics about unauthorized network access, use, and eavesdropping, mostly on open 802.11 based networks.

At the end of 2002, other organizations like Tenebris published the results of collecting data from a wireless honeypot [8] deployed in Ottawa (Canada). In 2003 [2] and 2004 [16], wireless honeypots were deployed around the city of London and Australia, to investigate the unauthorized use of wireless networks and to promote the idea of using wireless honeypots as a deception mechanism.

In 2004, Laurent Oudot [11] released the article "Wireless Honeypot Countermeasures". It was focused on providing an introduction to the goals, design and limitations of wireless honeypots, and provided practical examples using honeyd and FakeAP.

In 2006, Wireless Honeypot concept was used in the MAP project [26] (MAP - Measure, Analyze, Protect), in its Measurement component to develop a framework to address existing and future attacks on WiFi networks.

Recently in 2007, Raytheon, sponsored a wireless honeypot research project, dubbed "The Hive" [27], at the University of Florida, to help address wireless threats. The project is based on a Linux.

The research in this paper discusses the honeypot architectures focused on the deployment of wireless infrastructure networks, where an access point provides and controls the access to the medium. The honeypot models for other wireless technologies like Bluetooth [5] and peer-to-peer wireless networks [22], where the access to the medium is managed by one of the participants also exist. Compared to Wireless LAN, honeypot application in the world of Bluetooth and ad-hoc environment is still new and is evolving.

As discussed above, various wireless honeypot projects have been deployed worldwide to study the dangers involved in wireless networks. These projects has confirmed the huge war driving activity taking place at different parts of the world and that commercial world is in continuous threat. The aim of this paper is to create awareness about the current wireless threats and about the wireless honeypot technology which if deployed at large scale can help to reach at definitive conclusions about the wireless hacking scene and can improve the defences used to protect wireless networks.

For this, the paper first discusses the technologies used in some of the existing wireless honeypots systems/ infrastructures and the wireless tools that are used for creating virtual environment. Then, in Section 3, it presents a generic Wireless Honeypot Framework. This framework discusses the generic design requirements of a wireless honeypot and has been divided into six phases. In Section 4, it identifies different attack scenarios that are possible in wireless environment and studies the types of wireless honeypot architectures that would be needed to tackle each situation. In Section 5, it gives a comprehensive comparison of existing wireless honeypot systems and classifies the existing projects into one or more of the proposed architecture types. Finally through the observations made from the existing projects analysis, it throws light on advantages and limitations of existing approaches, and tells how a more comprehensive wireless honeypot (compared to previous systems) could be designed and implemented. The paper ends with Section 5's conclusions.

An important point to note is that honeypot is a deception technology. Like any other technology, it can be used for good or for bad. In one way, deception is used to lure the attackers and trap them. But recently it's been seen that attackers are also using AP faking the enterprise or legitimate hotspot AP's SSID to deceive and lure the legitimate clients and to launch denial of service, client mis-association and other attacks. Therefore to remove any confusion, we will use term Rogue Access Point for Evil Twins or AP deployed by attackers. And will continue to use term Honeypot for deception technology deployed by network administrator against the attackers.

2 Existing Wireless Honeypot Systems and Tools

2.1 WISE

WISE [13] was the first organized wireless honeypot designed to gather data about unwary Wi-Fi hackers and bandwidth borrowers, their techniques, attack signatures, frequency etc. WISE was an "802.11b network" deployed in Washington D.C. and was dedicated to no other purpose than being hacked from nearby. The system closely monitored all the activities that took place on the network.

The network had five Cisco access points, some deliberately vulnerable computers as bait, and two omni directional high-gain antennas for added reach to the nearby streets and alleys. On the back-end, a logging host was used to gather detailed connection data from the access points, while a passive 802.11b sniffer with a customized intrusion detection system was used to act as a hypersensitive trip wire. It had an Internet connection, hooked up through a web proxy that intercepted all outgoing connection attempts and presented a consent-to-monitor banner, to know about how Internet link was being used. Like conventional honeypots, the WISE network had no legitimate users, so anything that crossed it was closely scrutinized.

2.2 KPMG's Wireless Honeypot

KPMG- a London based consulting firm, set up a wireless honeypot in 2003 to lure London's wardriving commuters [9]. It was a dummy network that appeared as a legitimate corporate wireless network. Three separate

wireless points were set up at different points around the Square Mile in London, and were ran for a week each. The activities of all the users who try to access it were recorded and analysed. The aim of this whole set up was to establish the prevalence of wardrivers and wireless hackers.

An average of 3.4 'probes' was detected per working day. Most of the attackers did probing for fun, and in some case to use the network to access the Internet. The most popular time for war driving was between 9-10 am, where 24% of probes took place, and 5-6pm where 18% of probes took place. This suggested that people scanned for wireless access points while driving in cars, or while on foot or cycling. Virtually no activity was recorded at weekends. The wardrivers mostly tried to access wireless networks on the way to and from work.

Analysis of the probes also revealed that 84% of those looking for wireless networks simply identified the presence of the network and moved on. Such probes were expected to be for charting maps of wireless access points for future use. Sixteen percent of probes ended in eventual network access, and three-quarters of those who did access the network undertook hostile activities. The malicious behavior included attempts to access systems and tamper with their set-up, and attempts to run computer commands that would damage the technology.

2.3 Proactive WIDS

The concept of Proactive WIDS was introduced [7] in 2004. This system had 5 modules: Packet capture module, session analysis module, Intrusion detection, honeypot and alarm module. The attack detection was done by modules like packet capture, IDS and session analysis. When an attack was detected, Honeypot module was used to redirect the attacker into a fake AP and unnecessary information was given to the intruder. In this way the risk was shifted to non-production system. Honeypot quarantined all the attack events from the production IDS. Honeypot was also able to detect the jamming of management frames and to decrypt data frames on the fly and to re-inject them onto another device.

2.4 Deceptive Wireless Honeypot

Suen Yek [24, 25] implemented network defense using Deception-in-Depth (DiD) concept. It was a layered 3 ring model with each ring having different deceptive strength. The central core embraced the most effective deception and as the rings progress outward, the strength of the deception abated.

The peripheral, Ring 3, was the most vulnerable Fake AP layer. This ring produced an AP gateway for attackers to enter the ring 2. Using FakeAP software, one or many fake access points were simulated to confuse the attacker.

The Ring1, the inner most ring was the central logging structure encompassing the IDS SNORT acting as packet sniffer and Honeyd logs to passively record all system traffic. The central was the most important part of the model as it had all the network data like the source and destination: IP address, MAC address, TCP/IP ports and the protocols used, as well as any buffer outputs. This collected data was used to confirm network penetration.

2.5 HoneySpot

HoneySpot project [17] is based on attacks that try to break into a secure wireless network. Two types of HoneySpots have been defined - Public and Private HoneySpots.

A Public HoneySpot simulates a public wireless data network, that is, purehotspots networks available at hotels, airports, coffee shops, libraries, as well as other public places where there is a high interest in offering Internet connectivity to visitors and customers.HoneySpot for these networks don't have access control mechanism at the wireless level and focuses on wireless attacks at IP layer i.e. for "open" networks.

HoneySpot provides different levels for both the scenarios. For Public Honeyspot only one level is available Level 0 with Open wireless network (with IP-layer controls). For Private Honeyspot three levels are defined. Level 0 for WEP-based wireless network, Level 1 is a WPA-based wireless network and Level 2 is a WPA2-based wireless network. Their system has all the components- a Wireless Access Point Module, a wireless Client Module, a Wireless Monitor module, wireless data analysis Module, and an optional Wired Infrastructure module.

2.6 Wireless Tools

Attackers can be deceived of a real production network by deploying real wireless infrastructure resources (High Interaction Honeypot Technology) like real access points, real administrative serves which doesn't have any production work to do except monitoring the interaction done with and around them. Otherwise, one can deploy devices that emulate the behaviour of real resources (called Low Interaction Honeypots) for luring and deceiving the attackers.

Different architectures use different tools for deception. The two, low interaction, tools used in the above models - Honeyd and Fake AP are described here.

Honeyd: Honeyd is one of the most powerful, and most commonly used OpenSource honeypot developed and maintained by Niels Provos [6, 14]. In many ways, Honeyd is not a honeypot, but a honeypot toolkit, allowing one to build and customize the solution one wants. It can be configured and adjusted to emulate basic WLAN services and components in multiple ways. It can emulate a fake network routing topology on a wireless environment. It creates fake TCP/IP stacks to fool remote OS fingerprinting tools such as nmap or xprobe in combination with network

infrastructure and network routing emulation. This gives the appearance of actual wireless networks to an attacker. It can also emulate an AP by copying well-chosen web pages used to manage an access point. Attackers who try to compromise the Access points by using well-known default passwords at the management interface will get trapped by these fake web pages created by Honeyd. It can also monitor attackers who try to use opened services (such as attacks over SNMP, DNS, DHCP, TFTP, etc) by creating fake services.

However, like any other emulator, it also has its own limitations. Honeyd expects a specific type of behavior and it is programmed to react in a predetermined way. If attack A does this, then react this way. If attack B does this, then respond this way. Therefore, if the attacker does something that the emulation does not expect, then it does not know how to respond and simply generates an error message, breaking the deception.

Fake AP: Black Alchemy's Fake AP [4] generates thousands of fake 802.11b access points by manipulating the BSSID and ESSID fields. Listener sees thousands of fake access points. This tool can be use to confuse wardrivers, NetStumblers, Script Kiddies etc.

However, the idea behind this simple tool is very old. Today most updated tools can advise the attacker that the detected access points are fake and not real as no traffic is generated on the found networks.

3 Wireless Honeypot Framework

In this Section, a very Generic Wireless Honeypot Framework is detailed highlighting the essential modules of a wireless honeypot and their role in the overall network security. The whole process model of Wireless honeypots is divided into six phases, and the main considerations and requirements of each phase are given. (Refer to Figure 1).

Phase 1: Add and Remove Vulnerabilities
The first phase of setting up the trap is to deploy the bait/resources that will attract the attacker. It includes taking decisions regarding the production system we want to duplicate, the security policies of network and the level of interaction i.e. how much activity, or interaction, an attacker can have with the honeypot. This depends upon the type of attacks one wants to study and the level of risk the network can tolerate.

The quality of a honeypot lies in its ability to lure the attacker and to deceive the attacker of a real production system. Too many vulnerabilities reveal the identity of honeypot and too few vulnerabilities decrease its chances to be chosen by the attackers. Therefore in a calculated manner enough vulnerabilities are added to attract the attackers and at the same time some commonly known vulnerabilities are removed from system to give the feel of a real system.

Some attributes considered in this phase are:

1) Attack Scenario: Pure Layer-2 attacks/IP level attacks to target: specific attacks on web-based captive portal, or generic IP-level attacks common to wireless and wired environment.

2) 802.11 Technology used: 802.11 a/b/g/n.

3) Security Policy of network: Open (No authentication), WEP/WPA/WPA2 based Wireless Network, Encryption Method (None, TKIP, WEP, CCMP), MAC Address Filtering available or not, 802.1X/EAP type (PEAP, EAP, TTLS etc).

4) Infrastructure essentials/bait for the attackers: Wireless AP Module, Wired Infrastructure Module, Wireless Client Module, and Wireless Device Modules.

5) Level of Interaction: Emulated/Real infrastructure.

Network can have different levels of interaction at different modules. Attributes considered in different modules are:

1) Wireless AP module- Emulated/real AP and AP attributes like number of AP used.

2) Wired Infrastructure Module- Emulated/real wired infrastructure and the services (TCP, UDP, ICMP etc, and application layer services) exposed to the wireless network.

3) Wireless Client Module- Emulated/real wireless clients, number of wireless clients, type of wireless traffic between clients and AP, and client vulnerabilities available at the operating system level, at wireless administration client software, or at the wireless drivers.

4) Wireless Device Module: Emulated/real administrative servers.

One also needs to consider, during the design phase of Wi-Fi network, about how many Wi-Fi devices are already using the spectrum at that given area and which channels are busiest and which are unused yet in local area. This helps to plan the Wi-Fi network better and reduce interference with other Wi-Fi devices by choosing the least used channels for a new Wi-Fi network.

Phase 2: Monitoring and Logging Activities
The next phase is to monitor the activities of the trapped attacker without him knowing that he is being watched. This includes logging the interactions at different network modules- wireless APs, wired infrastructure, wireless client and wireless device administrative servers. The records of file changes, key strokes, services accessed etc are all logged.

Wireless sniffers are also deployed at different locations in the network to log on-air pre attack, during attack and after attack traffic. Softwares such as KisMAC or Kismet in combination with packet analyzers such as Wireshark or tcpdump provide user interfaces for passive wireless network monitoring. They hear in monitor (RFMON) mode which allows them to capture packets without being associated with an access point. This captured data in PCAP format includes information about how the attacker breaks the Layer-2 security policies. They can be configured to listen on the same wireless channel as the AP is configured or can have additional radios to constantly scan the activities taking place in the other channels. This gives information about the wardriving activities taking place around that area.

Phase 3: Creating Integrated Database
The next phase is to integrate/fuse the data logged and sniffed at different modules/locations of the network. This phase requires module with data fusion/integrating capabilities. Generally these capabilities are incorporated into the Data Analysis Module discussed in next phase. All the logged data - the PCAP files and System log files are collected for analysis at a remote location via wired or wireless connections.

Phase 4: Online Analysis
Some examples of rules: A Mac Spoofing detection rule can check for the first three bytes of Mac address which corresponds to its manufacturer. Wardriving activities can be identified by checking the existence of unique-ids. Specific unique-ids are present in packets generated by tools used for wardriving like Kismet and Netstumbler. Similarly, a WEP decryption attack by packet replay techniques can be detected by monitoring the probe response traffic level.

In this phase, according to the threat level detected, alerts are also generated in real-time to let the administrator know about the suspicious activities happening in the network and their behavior.

Phase 5: Investigation Phase
This is a post-event phase, the integrated/raw data of several weeks, including the signalling information (RSSI) and the results of online analysis are also sent to this phase for an offline analysis. This phase has Analysis Modules with advances forensics capabilities like tools which can triangulate the attacker's position based on the signal levels received and supporting advanced wireless-incidents handling tools.

This phase helps to find the locations from where the wireless attacks are performed relative to the wireless access point and wireless monitoring unit and to know about the tools (softwares, antennas and wireless card specifications) used by the attackers.

Phase 6: Modification Phase
This phase follows the results of various Analysis Phases.

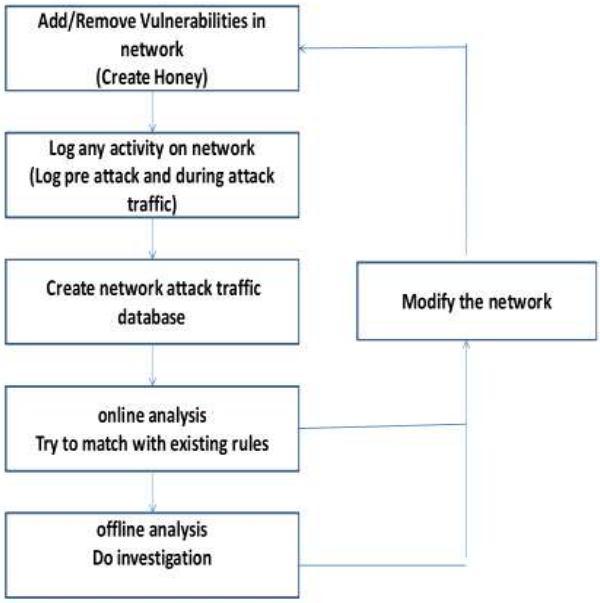

Figure 1: Wireless honeypot framework

Modifications in the network architecture and deceptive services are made depending upon the information obtained about weakness and limitations of the currently deployed architecture, and about the attackers and their new methodologies.

Modifications include incorporation of new rules generated by the online and offline analysis phases in the network's attack prevention and detection engines. Administrator can decide to increase the encryption level from WEP to WPA or WPA2. More robust username and password can be designed to avoid password guessing attacks. AP firmware can be upgraded to the latest version to avoid the exploitation of well-known vulnerabilities. Additionally, one could automate a procedure to reset the WAP and reinstate it to a known state periodically to ensure that its configuration has not been modified.

4 Architectures Based on Attack Scenario

In this section we have identified the different possible attack scenarios in wireless environment and have deduced separate wireless honeypot architecture to capture each scenario, using the currently existing techniques. The first architecture- Type A focuses on Layer-2 attackers that try to gain access to wireless network. The second architecture- Type B studies attackers who try to gain control over the wireless devices and try to change its configurations after getting access to the network. The third Type C is for the attackers who try to exploit the wireless clients connected to the network. Type D looks for attackers interested in wired infrastructure behind the

wireless access points. Type E is for monitoring all types of attackers. This section details the design requirements of architectures needed for luring such different types of attackers.

Type A: Honeypots for Layer-2 attacks

These honeypots looks for Layer-2 attackers that try to enter open/secured wireless networks using the information broadcasted by wireless access points.

The first step for any wireless attacker is to break the Layer-2 security level of the network. Access Point (AP) sends beacons in the air to tell about their presence. These beacons contain SSID (Service Set Identifier), time, capabilities, physical layer parameter sets and supported data rates. An attacker can exploit these features to launch an attack against the network. Layer-2 attacks include wardriving activities, WEP/WPA/WPA2 key cracking, packet replay attacks, Dictionary attacks, 802.1x/EAP attacks, Chop-Chop attacks etc. Entering an open network doesn't require any authentication. To break into a WEP/WPA/WPA2 based secured network sophisticated tools like airsnort and aircrack are needed. These tools capture the probe request packets in huge number from the access points in promiscuous/monitor mode and then apply decryption algorithms to crack the encryption key.

Honeypot create deception for such attackers by emulating the presence of too many fake access points in an area. Targeting one network is an easy task, but a cloud of targets confuses the attackers and make it difficult for them to attack the production network's AP. Fake access points (1-10000) can be emulated using FakeAP software. A single AP flicks from one single SSID to another thus appearing as multiple AP. It transmits fake beacons. This misleads the attacker and traps him into a virtual environment.

Honeypot create deception for such attackers by emulating the presence of too many fake access points in an area. Targeting one network is an easy task, but a cloud of targets confuses the attackers and make it difficult for them to attack the production network's AP. Fake access points (1-10000) can be emulated using FakeAP software. A single AP flicks from one single SSID to another thus appearing as multiple AP. It transmits fake beacons. This misleads the attacker and traps him into a virtual environment.

Honeypot deployed can also be a real AP with no production value. Honeypot's wireless monitoring devices like KisMAC or Kismet listening in monitor mode then logs the on-air traffic around the AP. This gives information about the techniques and tools used by the attacker to crack the 802.11 security standards (Open/Shared/PSK/Enterprise level authentications and RC4/AES type's encryptions). Figure 2 shows Honeypot architecture for Layer-2 attacks.

Type B: Honeypots for attacks against wireless infrastructure devices

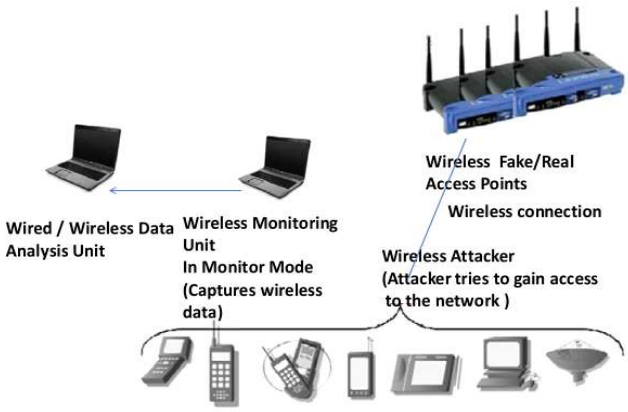

Figure 2: Honeypot based wireless network architectures for layer-2 attacks

These honeypots are directed against attackers who try to gain control of the AP or wireless controllers i.e. the wireless infrastructure devices.

Attackers changes the default/security specific AP configurations by gaining access to the management interface of the administrative server of AP using well-known default passwords, or through other opened services (such as attacks over SNMP, DNS, DHCP, TFTP, etc) at these servers.

To lure such attackers, Emulated/Real access point and administrative servers are deployed as shown in Figure 3. Honeyd tools are used to emulate wireless AP and administrative server services for attackers to connect to. It creates fake TCP/IP stacks to fool the remote fingerprinting tools such as Nmap or Xprobe. Fake web-servers, fake websites and other fake services are also created by honeyd to trap the attacker and to hide the real WAP controlling administrative server.

Type C: Honeypots for attacks directed at wireless clients

This architecture lures the attacker by deploying vulnerable wireless client as shown in Figure 4.

When a client connected to a wireless network is not well configured or is badly protected (such as laptop used from home and brought to a company), an attacker can exploit the vulnerabilities of one to attack the other. An attacker can deploy Rogue Access Point using tools like Karma [10], Wi-Fish Finder [3], and Hotspotter [23] which lures the clients attacker by sending stronger wireless signal than the official wireless AP and deceive them by pretending to be the official wireless AP and captures their credentials. They can then easily launch attacks like man in the middle attacks, denial of service, infection with a new worm that spreads itself on the rest of the legitimate network after the client reconnects itself, and so on. Some attacks against wireless clients are wireless driver vulnerabilities exploitation, Wireless client and driver fingerprinting, Wireless 802.11 protocols fuzzing, PSPF at-

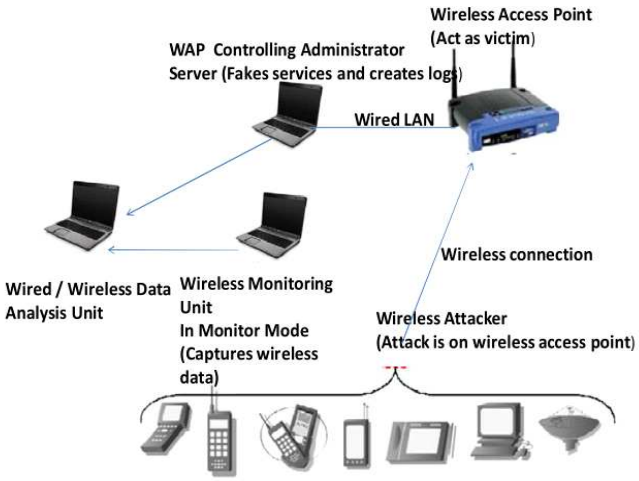

Figure 3: Honeypot based wireless network architectures for attacks against wireless access point

tacks (direct traffic injection and eavesdropping), Open and WEP-based spoofed access points, Preferred Network List (PNL) attacks and attacks on PEAP and TTLS configurations.

Honeypots in the form of vulnerable wireless clients are deployed. They have wireless cards configured in managed mode, with the appropriate settings to connect to the Wireless access point and are deployed at a standard distance from the access point. In honeypot emulations, client traffic is generated in such a way that a casual observer of the wireless network cannot easily determine that the traffic has been automated. To detect and deceive the Rogue Access points, honeypots with predesigned Preferred Network List (PNL) are deployed.

In honeypot emulations, client traffic is generated in such a way that a casual observer of the wireless network cannot easily determine that the traffic has been automated. Client Honeypot creates multiple individual connections with the wireless network, emulating the presence of multiple clients using different unique MAC addresses. Different traffic associated to one or multiple protocols (802.1x, ARP, ICMP, TCP, UDP, IPSec, etc) and applications, such as (secure) web browsing, FTP, SSH, VPN traffic, e-mail access, etc is generated at each connection. Traffic replay tools like tcpreplay, or traffic generators are used to generate customized traffic for individual connection. The different traffic profiles emulated are generated in random fashion and with varying information data exchanges.

Different Levels of complexities are added to the client honeypots as described in HoneySpot paper [17] based on the level of detail. Emulation of dormant clients is done by establishing a connection, exchanges some traffic, and then going back to sleep for a few minutes or hours. Such honeypots help to evaluate active and passive session of hijacking through MAC/IP address spoofing in hotspot-like environments.

Type D: Honeypots for attacks against wired infrastructure

This architecture looks for attackers who try to enter a wireless network to target the wired infrastructure connected to it (Refer to Figure 5).

Honeypots for such scenario are made by using a Real/Emulated Access Point plugged to a Real/Emulated wired network infrastructure. Honeyd can be used to emulate virtual LAN behind the wireless access points. It can set up a whole virtual Internet routing topology and huge fake wired networks behind the wireless environment.

Honeypot can also be made to emulate hotspot configurations. It can provide internet access to the users with free DNS traffic but requiring authentication for other services. Information about the remote IP of the servers, the attackers are trying to access during their unauthorized tunnel sessions, can be obtained in this way. But as suggested by Lance [12], while providing internet connections, it is important to block the outgoing network traffic using Intrusion Prevention System, like Snort-inline.

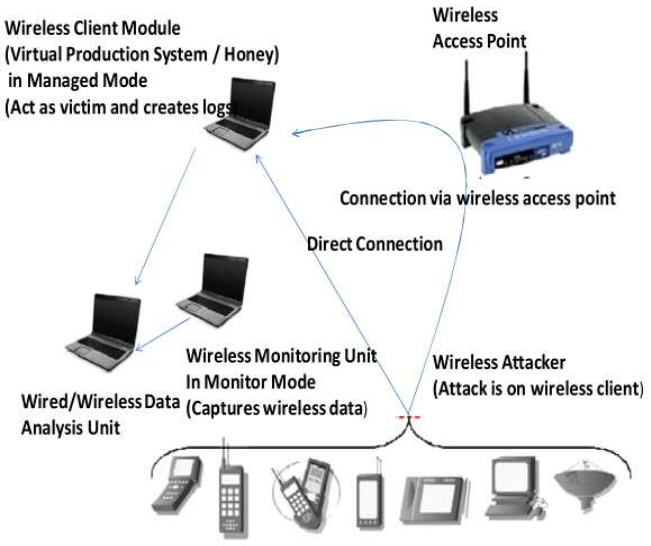

Figure 4: Honeypot based wireless network architectures for attacks against wireless clients

Type E: Integrated architecture

An integrated architecture covering all the attack scenarios discussed above can be made as shown in Figure 6. This architecture has wireless real/emulated access points to confuse the attacker of real target. There are faked web-interfaces for WAP administrative services. The wired environment behind the access point can also be an emulation to lure the attacker. There are wireless clients in the network to increase the realism of the wireless network and to emulate the wireless traffic for attackers to sniff. All different kinds of attackers can be monitored using this architecture.

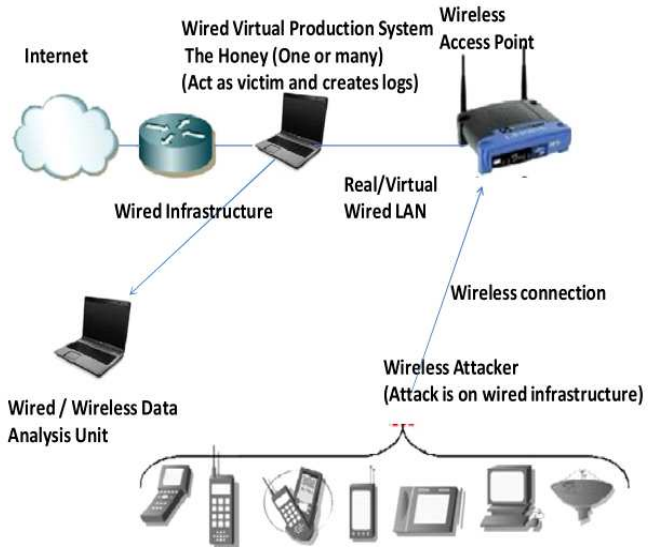

Figure 5: Honeypot based wireless network architectures for attacks against wired infrastructure

5 Analysis of Proposed and Existing Architectures

Table 1 compares and contrasts the existing honeypot systems (discussed in Section 2) on the basis of architectural types (discussed in Section 4), defense mechanism they use -detection, deception and deterrence mechanisms (explained in Introduction), the deployment technology used, their testing scenarios and results obtained from their research.

The observations made from the above analysis shows that wireless honeypot models of Type A, D, and E architectures have been proposed and deployed by researchers, but models focusing on only Type B and Type C architectures are not there. However, vulnerabilities of wireless clients and wireless devices' administrative servers can prove to be the weakest link in the network. Seeing the recently increasing threat in this domain, it is important to study these attack scenarios also in much detail by deploying Type B and C wireless honeypot architectures.

Another thing confirmed by the deployment of Honeypots is that secured networks (WEP/WPA/WPA2) are as much vulnerable to attacks as open networks. Much stronger encryption mechanisms are needed to enhance security level.

Most of these honeypot models follow either deception or detection or a combination of these network defense mechanisms. Moreover, they are mostly tested in simulated environment or are deployed for research purposes only.

Honeypots have tremendous potential for the security community, and they can accomplish goals few other technologies can. Therefore it is required that honeypots be deployed not only for evaluation of wireless attack scenario but also for providing protection to real production systems. Novel model, that integrates all the three- deception, detection and deterrence techniques, are needed to be deployed in industrial environment along with real production systems. As the deployment of different architectures will be increased worldwide, it is the deterrence that can be then exploited to control the overall attack scenario in wireless domain.

6 Conclusions

With the exponentially increasing usage of wireless devices, the threat of wireless attacks is also growing. This paper presents a comprehensive survey of previous wireless honeypot systems that have been deployed to assess the security scenario in wireless domain. Then, a framework is given which describes the whole process model of a generic wireless honeypot. The possible attack scenarios in wireless domain are identified and the honeypot architectures compliant with each scenario are given. A comprehensive analysis (classification and comparison) of existing projects is presented. As part of observations made from classification, the paper poses the need for worldwide deployment of more comprehensive wireless honeypots, incorporating all the three- deceptive, detection, and deterrence mechanisms, to control the wireless attack scenario.

References

[1] J. Chen, M. Jiang, and Y. Liu, "Wireless LAN security and IEEE 802.11i," *Wireless Communications*, 2005.

[2] P. Cracknell, *The Wireless Honeypot Project: A brief Look at How Wireless Networks Are Used and misused in the City of London*, Technical Report, RSA Security UK Limited (RSA SUL), CISSP, 2003.

[3] Dino A. Dai Zovi, Karma. (http://www.theta44.org /karma)

[4] B. A. Enterprises, FakeAP. (http://www.blackalchemy.to/project/fakeap/)

[5] A. Galante, A. Kokos, and S. Zanero, "BlueBat: Towards practical bluetooth honeypots," *Proceedings of the 2009 IEEE International conference on Communications*, pp. 1-6, 2009.

[6] Honeyd. (http://www.honeyd.org/concepts.php)

[7] W. C. Hsieh, C. C. Lo, J. C. Lee, and L. T. Huang, "The implementation of a proactive wireless intrusion detection system," *Proceedings of CIT '04*, 2004.

[8] E. Jacksch, *Tenebris Wireless Honeypot Project: Assessing the Threat Against Wireless Access Points*, CISSP, Tenebris Technologies Inc, 2002.

[9] KPMG, *Survey Reveals Hackers Hunt for Wireless Networks Whilst Commuting*, 27 Mar. 2003. (http://www.kpmg.co.uk/kpmg/uk/press/detail.cfm ?pr=1634)

Table 1: Analysis of existing wireless honeypot projects

	Architecture	**Deception/Detection System**	**Deployment Technology**	**Testing scenario**	**Results of Project**
WISE (2002)	Type A and Type D combination (Real Access Point and Real Wired Infrastructure)	Detection and Deception System: To gather data about the attacks on 802.11b open networks, the attack techniques, attack signatures, frequencies used etc.	It was an 802.11b open network with five real Cisco few access points, vulnerable computers and two omni directional on the back-end, it had a logging host, a passive 802.11b sniffer, an IDS and an Internet connection.	Real Time traffic was collected by deploying the set up at a secret location in Washington DC. Anything that crossed the network was closely scrutinized.	Simple war-driving activities were detected.
KPMG's Wireless Honeypot (2003)	Type A and Type D architecture (Real Access Point and Real Wired Infrastructure)	Detection and Deception System: To detect the prevalence of war-drivers and wireless hackers in London.	It was an 802.11b open network. Infrastructure had three real wireless access points. The set-up had dummy corporate wireless network at the back-end, with hidden recording and analysis units.	Real Time traffic was collected by deploying the set up at three different points around the Square Mile in London, and each was ran for a week.	An average of 3.4 'probes' was detected per working day. 84% of these were simple war-driving activities and 16% probes ended in network access and with three-quarters of them were malicious in nature.
Proactive WIDS (2004)	Type A and Type D combination (Fake Access Point and Virtual Wired Infrastructure)	Deception System: Honeypot module is used for deception only, protecting the real production system from war-driving, WEP cracking, Mac Spoofing and Telnet based attacks.	802.11 b secured network was deployed. The Honeypot Module consists of Fake AP and virtual Internet topology at the back-end.	Testing was done in simulated environment. Tools like AirSnort and Netstumbler were used for simulation of attacks.	System was effective in preventing WEP Cracking, Mac Spoofing, War-Driving and content violation attacks.
Deceptive Wireless Honeypot (2004)	Type A and Type D combination (Real Access Point and Real Wired Infrastructure)	Deception and Deception System: To protect the network from attacks on Access points and on wired infrastructure.	It was an 802.11 open network. Ring 3 was created using FakeAP software. Honeyd was used for creating Ring 2. The Ring 1 was a central logging structure encompassing the IDS SNORT packet sniffer and Honeyd logs.	Testing was done in simulated environment. The effectiveness of deceptions was evaluated from results gathered from the log-file/report outputs of four commonly used network attack tools: Kismet, Netstumbl, Network Mapper(NM-AP) and Nessus.	FakeAP was successful in deceiving wireless sniffing tools. The network-attacking tools were able to perceive through the Honeyd deceptions. So Honeyd deceptions were not much effective.
HoneySpot (2007)	Type E Integrated architecture	Detection and Deception system: To detect mostly layer 2 wireless attacks and IP layered attacks against wireless clients, wireless infrastructure and wired networks.	The system can be any of the 802.11a or 802.11b network. The network has all the modules: Wireless Access Point Module, a Wireless Client Module, a Wireless Monitor module, wireless data analysis Module, Infrastructure module. The Public Honeyspot is an open network with IP layer controls and Private Honeyspot is with open/shared/PSK/Enterprise level authentication methods and WEP/WPA/WPA2 technologies implemented.	The private type HoneySpot was tested in simulated environment for location in Layer-2 attacks.	The network was successful in detecting and deceiving in all Layer-2 attack scenarios.

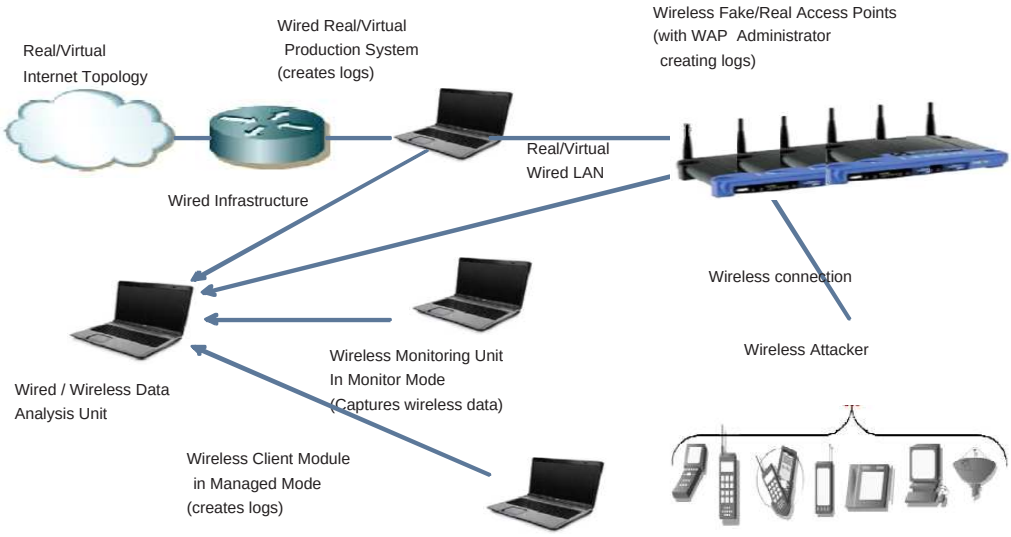

Figure 6: Honeypot based integrated wireless network architectures

[10] Max Moser, Hotspotter. (http://www.remote-exploit.org/)

[11] L. Oudot, *Wireless Honeypot Countermeasures*, Securityfocus, 2004. (http://www.securityfocus.com/infocus/1761)

[12] L. Oudot, *Wireless Honeypot Trickery*. (http://security.ittoolbox.com/documents/wireless-honeypot-trickery-17302)

[13] K. Poulsen, *W-Fi Honeypots a new Hacker's Trap*. (http://www.securityfocus.com/news/552)

[14] N. Provos, "A virtual honeypot framework," *Proceedings of USENIX Security Symposium*, 2004.

[15] P. Pudney, *An Investigation into the Unauthorised Use of 802.11 Wireless Local Area Networks*, Thesis, School of Computer and Information Science, University of South Australia, 2005.

[16] P. Pudney and J. Slay, "An investigation of unauthorised use of wireless networks in Adelaide," *ACISP 2005*, LNCS 3574, pp. 29-39, Springer-Verlag, 2005.

[17] R. Siles, *HoneySpot: The Wireless HoneyPot*, The Spanish Honeynet Project, Spain, Dec. 2007. (http://honeynet.org.es/papers/honeyspot/HoneySpot_20071217.pdf)

[18] L. Spitzner, *Honeypots: Definitions and Value of Honeypot*, 2002. (http://www.tracking-hackers.com/papers/honeypots.html)

[19] L. Spitzner, *Problems and Challenges with Honeypots*, 2004. (http://www.symantec.com/connect/articles/problems-and-challenges-honeypots)

[20] G. Urbas and T. Krone, "Mobile and wireless technologies: security and risk factors," *Trends & Issues in Crime and Criminal Justice*, vol. 329, 2006.

[21] Wireless Security, Wikipedia. (http://en.wikipedia.org/wiki/Wireless_security)

[22] J. White, J. Brown, S. Ramaswamy, S. Geogha, and M. Itmi, "Securing P2P wireless communications by deploying honeytokens in a cooperative maritime network," *Proceedings CISSE '09*, 2009.

[23] WiFish-Finder. (http://blog.airtightnetworks.com/tag/honeypot)

[24] S. Yek, "Implementing network defense using deception in a wireless honeypot," *2nd Australian Computer, Information and Network Forensics Conference*, Fremantle, Western Australia, 2004.

[25] S. Yek and W. Australia, "Measuring the effectiveness of deception in a wireless honeypot," *Proceedings 1 st Australian Computer Network and Information Forenscis Conference*, pp. 1-10, 2003.

[26] *The MAP Project*, Dartmouth College, 2006. (http://www.cs.dartmouth.edu/ map/)

[27] *Wireless honeypots*, Randall Brooks, Raytheon, 2007. (http://www.raytheon.com/technology_today/current/feature_5.html and http://www.cise.ufl.edu/class/thehive/)

Radhika Goel Integrated Dual Degree (B.Tech in Computer Science with M.Tech in Information Technology) final year student at the Department of Electronics and Computer Engineering IIT Roorkee, India. She is currently working on her Master's dissertation. Her areas of interest include Network Security, and Data Mining.

Anjali Saradana Phd in Electronics and Computer Engineering from IIT, Roorkee, India (2009). She is presently an Assistant Professor in the Department of Electronics and Computer Engineering IIT Roorkee, India. Areas of Interest include Information and Network Security, Wireless Security and Intrusion Detection. Dr. Sardana has been working in the area of security and honeypots for the past 7 years. She has delivered several expert lectures at faculty training programs, seminar and workshops, and has published articles in international journals. She has received several awards incuding Google Women in Engineering Award and Government of India R&D award.

R.C. Joshi Phd in Electronics and Computer Engineering from University of Roorkee, India(1980). Teaching for the past 42 years and is presently Professor in the Department of Electronics and Computer Engineering, IIT Roorkee, India. Dr. Joshi has been involved in wide spectrum of fields like Parallel and Distributed Processing, Data mining, Information Systems, Bioinformatics, Information Security and Digital Forensics. He has chaired several conferences and has delivered various special lectures. He was awarded the prestigious Gold Medal by Institute of Engineers (India) in 1978. He was also a member of National Industrial Research and Development Award Committee. He is also Chairman of Planning and Curriculum Development Ambedkar University, Lucknow and various MIT and AICTE committes.