Review on RAP: Protecting Wi-Fi Networks from Rogue Access Points

Prof. Pranit S. Thakur **Prof. Vijay B. Patil** **Prof. Abhijit S. Bodhe**

Abstract- In RAP, novel techniques are introduced to detect rouge APs and to improve network resilience. Here an efficient rogue AP protection system termed as RAP for commodity Wi-Fi networks is proposed. This system has the following nice properties: it requires neither specialized hardware nor modification to existing standards; the proposed mechanism can be integrated with an AP in a plugin manner; it provides a cost effective security enhancement to Wi-Fi networks by incorporating free but mature software tools; it can protect the network from adversaries capable of using customized equipment and violating the IEEE 802.11 standard.

INTRODUCTION

It is the prevention of unauthorized access or damage to computers using wireless networks. Wireless networks are very common, both for organizations and individuals. Many laptop computers have wireless cards pre-installed. The ability to enter a network while mobile has great benefits. However, wireless networking has many security issues. Crackers have found wireless networks relatively easy to break into, and even use wireless technology to crack into wired networks. As a result, it's very important that enterprises define effective wireless security policies that guard against unauthorized access to important resources. Wireless Intrusion Prevention Systems (WIPS) or Wireless Intrusion Detection Systems (WIDS) are commonly used to enforce wireless security policies [3].

One of the most challenging security concerns for network administrators is the presence of rogue wireless access points. A rogue access point (RAP) is a wireless access point that has either been installed on a secure company network without explicit authorization from a local network management or has been created to allow a cracker to conduct a man-in-the-middle attack.

The rogue access points are devices that are deployed in secure WLANs without permission or knowledge of the network administrator. The presence of such rogue access point poses severe threats to the WLAN security as it could compromise security of the entire wireless LAN. This problem has been in existence ever since WLANs have become popular in commercial applications. There have been reports of data theft, identity theft by using these rogue access points. Increasing use of wireless technologies by defense establishments along with above mentioned reasons have compelled researchers all over the world to find a solution for this problem. WLANs face the same security challenges as their wired counterparts, and more.

Within a properly secured WLAN, rogue access points are more damaging than rogue users. Unauthorized users trying to access a WLAN likely will not be successful at reaching valuable corporate resources if effective authentication mechanisms are in place. Major issues arise, however, when an employee or hacker plugs in a rogue access point. The rogue access point helps an attacker in gaining access to sensitive information of an organization.

Employees have relatively free access to a company's facilities, which makes it possible for them to inadvertently (or mischievously) install a rogue access point. An employee, for example, installs his personal access point without permission of network administrator in order to support wireless printing or access to the network from a conference room. Software programmers working on wireless applications may connect an access point to the corporate network for testing purposes.

In order to avoid this situation, it is necessary to implement security policies that mandate conformance with effective security controls and coordination with the network administrator before installing access points. This can only be effective, nonetheless, if you clearly inform employees of the policies. After performing several security audits, it has been found that employees often install rogue access points without knowing the company security policies or the consequences of violating the guidelines. A hacker can install a rogue access point to provide an open, non-secure interface to a corporate network. In order to do this, the hacker must directly connect the access point to an active network port within the facility. This requires the hacker to pass through physical security; however, that's easy to do in most companies. Therefore there is an urgent need of developing technology which will address this problem of rogue access points.

With the increasing popularity of Wi-Fi networks, securing such a network becomes a challenging problem. Commodity Wi-Fi networks are particularly vulnerable to attacks because of factors such as open medium, insufficient software implementations, potential for hardware deficits, and improper configurations. Among all the security threats, one of the most dangerous hazards is the prevalence of rogue APs. A rogue AP is typically referred to as an unauthorized AP in the literature. This type of device can be easily deployed by end-users. When a rogue AP is connected to a network, it can be used by adversaries for committing espionage and launching attacks. Similarly, improperly configured APs and phishing APs can introduce the same security threats once exploited by adversaries. Therefore, they can be regarded as rogue Aps as well. More importantly, there is a more insidious type of rogue APs,

called the compromised APs, that has drawn little attention in the literature.

A compromised AP is the most dangerous rogue AP that can exist in commodity Wi-Fi Networks. In particular, it is difficult to detect such a rogue device because the AP itself is not malfunctioning (e.g., operating without specified security controls).

Table 1. Taxonomy of rogue APs.

Rough AP Class	Possible Scenarios
Improperly Configured	Insufficient security knowledge; faulty driver; physically defective; multiple network cards
Unauthorized	connected to internal LAN without permission; external neighborhood AP
Phishing	fabricated by adversary
Compromised	disclosure of security credentials

Further, the AP does not display anomalous misbehavior such as broadcasting a duplicate SSID. Thus, a compromised AP can significantly diminish the overall security of the network. A summary of the types of rogue APs and a number of possible scenarios is shown in Table 1.1 for a detailed taxonomy of rogue APs.

We first give a comprehensive taxonomy of rouge access points (APs), which includes a new class of rouge APs never addressed in the literature before.

RELATED WORK

Due to the security threats that a rogue AP can pose for corporate Wi-Fi networks, detecting such APs is one of the most important tasks of an IT department. Traditional rogue AP detection relies on network enumeration tools (e.g., NetStumbler) running on laptops or handheld devices carried by IT personnel. This “walking audit” approach is both time-consuming and unreliable. Further it fails when a rogue AP spoofs characteristics such as the MAC address and Service Set Identifier (SSID) of a legitimate AP.

To help automate the scanning process and provide continuous monitoring capabilities, a number of commercial products have been developed [2–4]. AirDefense [2] is one such product. It uses a combination of radio frequency sensors and a IDS/IPS server appliance to capture, process, and correlate network events. However, the latest release, AirDefense 7.2, has a starting price of US $7, 995. Lastly, if the specialized monitoring sensors are not used, it is difficult to guarantee complete coverage of the network to ensure effective rogue AP detection. On the other hand, the research community has just recently started to direct attention toward rogue AP detection. Architecture for fault diagnostics in IEEE 802.11 networks is presented in [12].

Multiple APs and mobile clients perform RFmonitoring to help detect the presence of rogue wireless devices like unauthorized APs. Each client is required to install special diagnostic software, and rogue APs are assumed to transmit beacon messages and respond to probe requests. In contrast, RAP does not inconvenience clients with additional software installs. Further, its detection ability is not based on the assumption that rogue APs will function properly.

EXISTING AP DETECTION SOFTWARES

1. NetStumbler

NetStumbler (also known as Network Stumbler) 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:

- Wardriving
- 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

As of February 2010, the author is working on an updated version that will work correctly with Windows Vista and Windows 7

2. MiniStumbler

A smaller version of NetStumbler designed to work on PocketPC 3.0 and PocketPC 2002 platforms. It provides support for ARM, MIPS and SH3 CPU types.

3. WEPCrack

WEPCrack was the first of the WEP encryption cracking utilities. WEPCrack is an open-source tool used to break 802.11 WEP keys.

4. Airsnort

Airsnort is a wireless LAN (WLAN) tool which cracksEP encryption keys. AirSnort passively monitors wireless transmissions and automatically computes the encryption key when enough packets have been gathered. AirSnort is an even simpler program, as it is completely interface-based. As the attack is only a simple brute-force attack however, cracking the encryption can take a while (from several days to a few weeks). Especially if traffic is low (only 4 users or so on network, the cracking will take at least 2 weeks).

5. BTScanner

BTScanner allows you to extract as much information as possible from a Bluetooth device without the requirement to pair. It extracts HCI and SDP information, and maintains an open connection to monitor the RSSI and link quality.

6. FakeAP

The polar opposite of hiding your network by disabling SSID broadcasts- Black Alchemy's Fake AP generates thousands of counterfeit 802.11b access points. As part of a honeypot or as an instrument of your site security plan, Fake AP confuses Wardrivers, NetStumblers, Script Kiddies, and other scanners

7. Kismet

Kismet is an 802.11 wireless network detector, sniffer, and intrusion detection system. Kismet identifies networks by passively collecting packets and detecting standard

named networks, detecting hidden networks, and inferring the presence of non-beaconing networks via data traffic.

8. Redfang

Redfang v2.5 is an enhanced version from @Stake of the original Redfang application that finds non-discoverable Bluetooth devices by brute-forcing the last six bytes of the device's Bluetooth address and doing a Read_remote_name().

9. SSID Sniff

A tool to use when looking to discover access points and save captured traffic. Comes with a configured script and supports Cisco Aironet and random prism2 based cards.

10. Wi-Fi Scanner

Wi-Fi Scanner analyzes traffic and detects 802.11b stations and access points. It can listen alternatively on all 14 channels; write packet information in real time, search access points and associated client stations. All network traffic may be saved in the libpcap format for post analysis.

11. wIDS

wIDS is a wireless IDS. It detects the jamming of management frames and could be used as a wireless honeypot. Data frames can also be decrypted on the fly and re-injected onto another device.

12. WIDZ

WIDZ is a proof of concept IDS system for 802.11 wireless networks. It guards access points (AP's) and monitors local frequencies for malicious activity. It detects scans, association floods, and bogus/Rogue AP's. It can also be integrated with SNORT or Real Secure.

Table 2 Existing AP detection softwares

Software	Network Monitor	Network Analyser	Encryption	Wired	Wireless
NetStumbler	Yes				Yes
MiniStumbler	Yes				Yes
WEPCrack			Yes		Yes
Airsnort	Yes	Yes			Yes
BTScanner			Yes		Yes
FakeAP		Yes			Yes
Kismet				Yes	
Redfang	Yes			Yes	
SSID Sniff			Yes		Yes
WiFi Scanner		Yes	Yes		Yes

THE RAP SYSTEM

1. The RAP system

RAP is designed to monitor network activities, forestall events that could lead to the generation of rogue APs, block unauthorized network access through rogue APs, and eliminate existing rogue APs. The three main components that constitute the RAP architecture are: a packet collector, a rogue AP preemption engine, and a rogue AP detection engine. An illustration of the overall architecture of RAP can be seen in Fig 1. The packet collector is responsible for gathering wireless traffic. The collected data is then passed to the preemption engine, where checks are performed in order to thwart various attacks. Finally, the data is analyzed by the detection engine. There are also probing functions shared by the preemption and detection engines so that adversaries can be lured into revealing their presence. These components can be implemented on an AP or on separate devices that connect to the AP in a plugin manner [6]. It is important to consider network performance when making the above decision. On a resource constrained AP, the overall network service could be degraded when all three components are implemented on it. The details of each component are described in the following subsections.

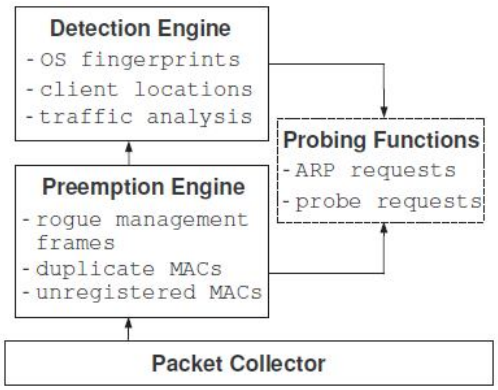

Figure 1. The RAP System

1.1 Assumptions

The type of Wi-Fi network that we consider uses WEP or WPA in conjunction with MAC address filtering. This combination of features is prevalent in commodity Wi-Fi networks. Additionally, we try to avoid rekeying activities, as they require significant overhead. An example of such a network is the one used by the Department of Computer Science at The George Washington University. Although there are about 20 to 30 active users daily, there are over 600 registered users.

Some research argues that monitoring a network from devices such as APs cannot provide comprehensive coverage. Yet, the coverage problem can be solved with the participation of multiple APs or the utilization of standard wireless range extenders. These extenders can be purchased for less than US $80 from online retailers. Based on the specifications of an off-the-shelf extender (e.g., the Belkin F5D7132), these types of devices can scan all working channels of 802.11b and 802.11g. Additionally, they offer a working range up to 457.2 meters. We assume that a floor plan of the building containing the to-beprotected network is available to RAP. In particular, the exact location of authorized APs and range extenders should be known by RAP. The above location data, along with information such as an AP's MAC address, SSID, nearest extender, working channel, and typical Received Signal Strength Indicator (RSSI) can be made accessible to RAP.

1.2 Packet Collector

A packet collector is needed for real-time WLAN monitoring so that rogue wireless devices can be quickly

identified, and network administrators notified when appropriate. One benefit of a packet collector is its natural ability to separate wired and wireless traffic. Thus, there is no need for complicated modules that attempt to isolate the two by examining traffic signatures.

1.2.1 Information Collection

The packet collector needs to have a network device that runs in promiscuous mode at all times. The entire region of interest can be covered with the assistance of wireless range extenders. One of the packet collector's duties is to capture all network traffic. In order to measure the storage overhead, we conducted a test using our department (802.11b/g compatible) Wi-Fi network. The test consisted of a single 802.11b client using ftp to download a large file from a local server. We carried out the test in the middle of the night so that there would be no contention from other wireless clients. The average transfer rate recorded over many trials was approximately 2.2 Mbps. Therefore, by recycling the collected data every one hour, the storage overhead can be limited to about 1 Gigabyte. This is a reasonable overhead for even low-end computing equipment.

Additionally, the packet collector dissects frames into IP and TCP components. This allows for information such as client MAC addresses, SSID, channel assignment, encryption status, and beacon interval to be recorded. It also filters the collected traffic into user specific streams such as AP-client pairs. The relevant data will be processed by the intrusion preemption and detection engines described in the following subsections. Lastly, the measured RSSI values from both APs and extenders are provided to the detection engine. Note that it is important to hide the packet collector from adversaries. Otherwise, a prudent attacker may change tactics to elude capture. To prevent inadvertently revealing its presence, RAP performs all sniffing in a passive manner. However, achieving real passive is not trivial. There is a well-accepted myth concerning passive listening: a network card in promiscuous mode and a properly configured firewall cannot reveal the presence of an eavesdropping device. In actuality, there are a number of ways that a "passive listener" can actually be active. Since firewalls filter at the IP layer and above (e.g., at the transport layer to keep state), not all traffic can be blocked. Examples of protocols that generate such traffic include:

1) The Address Resolution Protocol (ARP), a protocol that is primarily used for translating IPv4 addresses to Ethernet MAC addresses;

2) An extension to ARP called Inverse Address Resolution Protocol (InARP) that performs the inverse of ARP in Frame Relay networks;

3) The Bootstrap Protocol (BOOTP), which allows a client to obtain an IP address automatically during the bootstrap process 1 in the following, we propose techniques that achieve real "passive listening." We detail these techniques below.

1.2.2 Methods to Achieve Passive Listening

The first technique we describe involves disabling some options on a network card, while the second relies on a minor modification to the source code of the TCP/IP stack. Reconfigure Network Card: One way to prevent responding to any message that uses IP related information2 is to turn off the TCP/IP stack on a wireless card. This can be done by bringing up the card with no IP address configuration. Since there is no IP information available, no IP packets can be sent. However, the network card, as a device, can still be controlled by the OS for collecting frames in the air. If IP functionalities are desired, the network device can be restarted after the configuration file is changed back. Recompile TCP/IP Stack: In a wired network, an eavesdropper could physically cut the transmit wires in a network cable to ensure that messages are never sent. Although the above technique will not work in a wireless environment, modifying the source code of the TCP/IP stack can produce a similar effect.

The "send()" function in the TCP/IP stack is responsible for sending packets. By simply disabling the "send()" function and recompiling the stack, it will no longer be able to transmit packets. Once the source code has been modified, the new TCP/IP stack can be reloaded into the kernel. We note that recompiling a TCP/IP stack might prove to be a time consuming task. Additionally, it could be an inconvenience in environments where frequent changes in networking functionality are needed. Other network protocol stacks may also reveal the presence of a network device. One such protocol stack is Internetwork Packet Exchange/Sequence Packet Exchange (IPX/SPX), which is used by the Novell Netware operating system. IPX and SPX are the counterparts to the IP and TCP layers, and IPX can also be transmitted over Ethernet [1]. Similarly, the Network Basic Input/output System (NetBIOS) protocol can reveal network presence. This protocol assigns each computer on a LAN a NetBIOS name and an IP address in order to allow applications on separate computers to communicate.

Consequently, a network device, such as a packet collector, needs to have the above protocols disabled as well. 1Although this protocol operates at the IP/Transport layer, it can reveal an attacker's realMAC address before a spoofed address can be set.

1.3 Rogue AP Preemption Engine

While attempted network attacks cannot be avoided, it is possible to prevent some attacks before they happen. In particular, a certain amount of information must be collected by an adversary before an attack can actually occur. The prompt identification of such activity can help thwart an impending attack. Subsequently, a rogue AP preemption engine is included with RAP. The rogue AP preemption engine of RAP is our first line of defense.

The basic objectives of this component are to trap sniffers and thwart activity that can lead to AP compromise. Probing of potential eavesdroppers and network integrity checks are performed to accomplish these goals. The former is designed to discover passive listeners while the latter is used to prevent a legitimate AP from being compromised.

1.4 Eavesdropper Probing

Probing functionality is employed to help prevent Class 4 rogues from appearing on a network. In particular, messages are periodically generated that, when replied to, reveal the presence of a sniffer. One type of message is an ARP request. If a potential attacker is "passively listening" to the network traffic and replies to one of the trap ARP requests, her presence will be revealed. The interval selected for broadcasting these frames reflects a tradeoff between available bandwidth consumption and time needed for detection. These parameters can be customized based on the capabilities of the underlying systems.

1.5 Attack Preemption

After obtaining data from the packet collector, the rogue AP preemption engine will perform the checks outlined below. By preempting attacks that could reveal the secret network key, we prevent the creation of Class 4 rogue APs.

1) Unregistered MAC addresses are temporarily stored together with their location information. This is because an attacker might disclose itsMAC address to the AP before the knowledge of a legitimate MAC address is acquired. The location information can be obtained by localization schemes proposed in the literature (e.g., [20]). Typically, such localization schemes require 3 to 4 basestations (or APs in our case) with known locations. This requirement is easily satisfied in commodityWi-Fi networks. Additionally, such information will be shared with the rogue AP detection engine described in Section 2.7.

2) Duplicate MAC addresses are temporarily removed from the MAC filter so that network access is denied. This can happen when an attacker spoofs aMAC address to that of a client that is currently connected. The location of any station using this MAC address will be made available by the APs. If one of the locations matches that of a previously unregistered MAC address, the location of the attacker is identified [9].

3) The presence of management frames (e.g., DE authentication frames) will be observed because many active attacks rely on the transmission of forged frames. Although it has been suggested in that management frames in 802.11i be authenticated, the WEP and WPA protocols do not support this functionality. Thus, the preemption engine needs to keep a record of all management frames that the AP sends out. By doing this, the transmission of a spoofed management frame to a client can be detected, and the AP can choose not to respond to requests from that particular client. For example, in order to launch a dictionary attack on the shared key used in a WPA-PSK enabled network, an attacker needs to capture the four authentication frames exchanged between a client and the AP [7]. To do this, an attacker may send out a spoofed de-authentication message to a client to force the client to re-authenticate to the AP. In this case, the AP refuses to perform the authentication process with the client. Thus, the attacker is prevented from capturing the frames needed to launch a brute-force attack on the key.

As a complement to the above three tactics, a warning message can be sent to the system administrator whenever a

spoofed MAC address or a forged management frame is detected. Nevertheless, there are some cases where an attacker might go unnoticed by our preemption system. For example, the attacker might choose to employ the passive listening techniques described in above Section. The attacker could also track legitimate MAC addresses for use at a later time. Once the attacker has acquired the secret key, the MAC address of a legitimate but currently not present client can be used. Since these types of activities may go unnoticed by our integrity check module, RAP includes the rogue AP detection capabilities described in the next subsection.

1.6 Rogue AP Detection Engine

There are two primary reasons for the rogue AP detection engine. First, defending against Class 1 − 3 rogue APs is an inherently reactive process. For example, there is no way to prevent an attacker from setting up a phishing AP outside of a private organization. Secondly, a sophisticated adversary may be able to evade the preemption techniques for Class 4 rouge APs. The rogue AP detection engine is responsible for discovering rogue APs regardless of what class they belong to. For Classes 1−3, the AP probing technique described in Section below is used to lure rogue APs into revealing their presence. Class 4 rogue APs are detected by first identifying traffic from an unauthorized user. Additional mechanisms are included for handling adversaries that are strong enough to use hardware that violates the 802.11 standard. We detail the steps used by RAP's detection engine below. An AP advertises its presence several times per second by broadcasting special frames that carry its SSID called beacons. Stations can discover an AP by passively listening for beacons, or by transmitting a probe request message to actively search for an AP with a specified SSID. Our detection engine uses active honeypot functionality3 to discover rogue APs by sending out probe requests. It is capable of detecting the first three classes of rogue APs. There is a common misconception that disabling the "Broadcast SSID" feature hides the SSID. In reality, disabling this feature only makes the AP transmit a null (zero-length) SSID in beacon frames and probe responses instead of the its actual SSID. There are still several other frames (e.g., probe requests, association requests, and reassociation requests) that carry the SSID. Hence, it is impossible to keep an SSID value secret without manually reconfiguring device drivers or hardware to violate the 802.11 standard. Therefore, a particular AP can be discovered from its probe responses. The next step is to determine whether or not it is a rogue

3 Examples of active honeypot systems include Strider HoneyMonkeys and the Honeyclient Project AP.

One way to do this is to compare the discovered APs with those belonging to a list of authorized APs. Any AP that is detected and does not appear in the authorization list can be labeled as a rogue device. The relevant values associated with each AP in the table of authorized APs include its

MAC address, SSID, working channel, and equipment vendor. Accordingly, our detection system has a probe request frame periodically sent out on all of the channels (e.g., 11 channels in 802.11b). This property increases the likelihood of a rogue AP being detected because any AP that hears the request will send a probe response back to RAP. In this response, information such as the MAC address must be included, even though the SSID may not be present.

If the reported MAC address matches an unregistered MAC address found during an integrity check, we can conclude that it belongs to a rogue AP. Finally, RAP can have the switch port that is associated with the rogue AP's MAC address closed to eliminate it from the network. In the event that a rogue AP spoofs a legitimate AP's MAC address and SSID, location information should be used to make a judgment. If an AP announces a legitimate MAC address, but has localization results that are inconsistent with those in the APMACto- location table, it can be considered to be a rogue AP. RAP also handles extreme cases where rogue APs have had their driver and/or firmware modified in such a way that neither beaconframes nor probe response frames are transmitted. As a result, there is no MAC address information available to draw a conclusion.

Nevertheless, a disassociation message can be sent to a client of the suspect AP based on the stream information collected by the packet collector. When the client sends out a reassociation request, the MAC address and SSID of the AP will be disclosed. Note that the above technique can thwart an adversary with a levelof strength that has never been assumed before. In particular, other work such as and (assume that an attacker does not have the ability to violate characteristics of the 802.11 standard). Although this assumption is reasonable in many cases, the protection of any system based on it can be undermined. RAP does not place this limitation on the capabilities of the adversary. Hence, it is able to provide both robust and comprehensive protection from rogue APs.

1.7 Compromised AP Detection

A compromised AP is detected by identifying an unauthorized client who is connecting to it. The client is detected by employing a combination of MAC address information and OS fingerprinting techniques. Recall that MAC address filtering is used by the networks under the protection of RAP. The first 3 bytes of a MAC address, the Organizationally Unique Identifier (OUI), allow us to determine the assignee of a particular card company. Therefore, it is possible to link a card's OUI with the OS on the laptop that uses the card. For example, cards with an OUI of 00 − 17 − F2 are known to be used by Apple Inc. in their MacBook line of computers. Alternatively, information about OS preference can be obtained when users register with the system administrator. A table mapping each MAC address to the OS preference can be created. These options reflect a tradeoff between potentially increased detection accuracy and overall system complexity.

The OS that is actually running on a suspect client can be identified with OS fingerprinting tools. Examples of active and passive OS fingerprinting tools are Nmap and p0f, respectively. With this information, we can identify potential attackers by looking for inconsistencies between the card manufacturer/preferred OS and the fingerprinting results. A discrepancy may be cause for a "red flag" to be generated about a particular client. We make a note of the following caveat. There are some rare cases where the above OUI-to-OS mapping should not be applied. For instance, a registered MAC address could belong to a PCMCIA card that is used by different laptops. If these computers are running different operating systems, false positives could be generated.

Similarly, there are some cases where a card is used by a machine that can boot into multiple operating systems. To reduce the impact of the above scenarios, our mapping based on OS preference can be used. It is also possible for a sophisticated attacker to defeat OS fingerprinting tools by modifying the characteristics of the TCP/IP traffic (e.g., ISNs, initial window sizes, and options) that they base their identifications on. A freely available tool that performs such functions is IP Personality. Still, there is no guarantee that an attacker knows the OS that typically runs on the machine with (the network adapter with) the MAC address she is spoofing. Then, a guess should be made as to what OS a client is using based on OS market share information. If the guess is correct, the attacker cannot be identified. In this case, intrusion detection based on techniques from machine learning and data mining could be performed.

For example, a profile could be created for each client that indicates their Internet usage characteristics. The aim of this exercise is to learn how one famous algorithm for constructing decision trees, ID3, works. You will do this by building a decision tree by hand for a small dataset. At the end of this exercise you should understand how ID3 constructs a decision tree using the concept of Information Gain. You will be able to use the decision tree you create to make a decision about new data.

CONCLUSION

In this paper, we provided a comprehensive taxonomy of rogue APs. Our classification includes improperly configured APs, phishing APs, unauthorized APs, and a new class of rogue AP termed as compromised APs. This technique, when used in conjunction with an allowed AP policy or access list, can easily identify rogues. Also, this solution will function independent of the signal range of the rogue APs. We then develop a novel system for protecting commodity Wi-Fi networks from rogue APs called RAP. An attractive feature of RAP is that it requires neither specialized hardware nor modification to existing security standards. Further, the proposed mechanism can be connected to or implemented on APs as small plugins. It also makes use of freely available mature software in order to provide a cost-effective security solution. Lastly, RAP can protect networks from rogue APs even when assuming

that adversaries have the ability to use customized equipment that violates the IEEE 802.11 standard. RAP is the first system that can successfully protect the network under that assumption. As a part of our future work, we plan to deploy RAP on a test Wi- Fi network. Additionally, we are anticipating the inclusion of new features for RAP that can further improve its network protection abilities. One such feature is a proactive honeypot module that can be used to better preempt various attacks.

REFERENCES

[1] Liran Ma Department of Computer Science The George Washington University Washington, DC 20052, USA lrma@gwu.eduRAP: Protecting Commodity Wi-Fi Networks from Rogue Access Points
[2] M. A. Maloof. Machine Learning and Data Mining for Computer Security: Methods and Applications (Advanced Information and Knowledge Processing). Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.
[3] P. Mateti. Hacking techniques in wireless networks.
[4] M. Raya, J.-P. Hubaux, and I. Aad. Domino: a system to detect greedy behavior in ieee 802.11 hotspots. In MobiSys '04, pages 84–97. ACM Press, 2004.
[5] D. Schwab and R. Bunt. Characterising the use of a campus wireless network. In INFOCOM, 2004.
[6] Y.-M. Wang, D. Beck, X. Jiang, R. Roussev, C. Verbowski, S. Chen, and S. T. King. Automated web patrol with strider honeymonkeys: Finding web sites that exploit browser vulnerabilities. In NDSS, 2006.
[7] J. Yeo, M. Youssef, and A. Agrawala. A framework for wireless lan monitoring and its applications. In WiSe '04, pages 70–79. ACM Press, 2004.
[8] Nmap: Network mapper.
[9] p0f: a versatile passive os fingerprinting tool.
[10] A. Adya, P. Bahl, R. Chandra, and L. Qiu. Architecture and techniques for diagnosing faults in ieee 802.11 infrastructure networks. In *MobiCom '04*, pages 30–44. ACM Press, 2004.
[11] P. Bahl, R. Chandra, J. Padhye, L. Ravindranath, M. Singh, A. Wolman, and B. Zill. Enhancing the security of corporate wi-fi networks using dair. In *MobiSys '06*, pages 1–14. ACM Press, 2006.
[12] A. Balachandran, G. M. Voelker, P. Bahl, and P. V. Rangan. Characterizing user behavior and network performance in a public wireless lan. In *SIGMETRICS '02*, pages 195–205. ACM Press, 2002.

AUTHOR'S PROFILE :- 1. PROF. PRANIT S. THAKUR

Name:PranitShriniwas Thakur
E-mail:pranitbhai@rediffmail.com **Mobile No.: +91-9975772212**

WORK EXPERIENCE

Total Experience: 2 Year

Projects(3+ months):	
1.Cryptography	
Location	JES's S.N.D. Polytechnic, Babhulgaon- Yeola(Nasik).
Programming Language	Java
Team Size	4
Technologies	• Java Swing • Java Standard libraries
Role	• Study of functional requirements • Guidance for Designing of major modules and testing.
2.Image Steganography	
Location	JES's S.N.D. Polytechnic, Babhulgaon- Yeola(Nasik).
Programming Language	Java
Team Size	4
Technologies	• Java Swing • Java Standard libraries
Role	• Study of functional requirements • Guidance for Designing of major modules and testing.

Teaching Experience(9 month):	
College	JES's S.N.D. Polytechnic, Babhulgaon- Yeola(Nasik).
Date of joining	1st January, 2010 – 1st September, 2010.
Post	Lecturer
Achievements	Chief Co-ordinator of Event 'KHITIJ 2010'.
Subjects	• C programming - Practical (Result 97%) • Computer Graphics (Result 95%) • OOPL – Practical (Result 96.6%) • JPR(Java Programing)
Extras	• Main Project Co-ordinator. • Project guide.

Teaching Experience(1.5 year,till date):	
College	JES's S.N.D. C.O.E. & R.C., Babhulgaon- Yeola(Nasik).
Date of joining	2nd September, 2010.
Post	Lecturer.
Achievements	Departmental NBA Co-ordinator.
Subjects	• Computer Organization. • Computer Graphics. • OOPCG – Practical. • JPR(Java Programing)
Extras	• Seminar Co-ordinator. • Project guide.

Educational Qualification:

Degree	College	University/ Board	Percentage	Class	Year
M.E.(CSE)	MIT, Aurangabad	BAMU	Appeared	-	2010-11
B.E.(Comp)	SRESCOE, Kopargaon	Pune	62.13%	**First Class**	**2008-09**
T.E.	SRESCOE, Kopargaon	Pune	60%	**First Class**	**2007-08**
H.S.C.	J.V. Yeola	Maharashtra	59.33%	**First Class**	**2002-03**
S.S.C.	J.V. Yeola	Maharashtra	73.46%	**First Class**	**2000-01**

Software Exposures:	
Operating Systems	Ms-Dos, **Win95/98/2K/Xp/Vista/7/Azure,Unix**
Languages	*C, C++, Assembly language, Visual basics6.0, Core Java(SSI Certified,78%).*
Technologies	*J2SE*
Database Tools	*Oracle, SQL*
Web Technologies	*HTML*
Office Productivity Tool	Ms-Office-2003, Ms-Office-2007.

PROJECTS

BE Project	
Title	Video Stegnography (National Level 2nd Price Winner)
Location	**SRESCOE Kopargaon**
Team Size	4
Technologies	Java swing, RXTA tool
Role	• Study of functional requirements • Designing of major modules and testing. • As Developer developing major part of Project.

SEMINAR

TE Seminar	
Title	Next Generation of intel IXP Network Processors
Location	**SRESCOE Kopargaon**

MINIPROJECTS

TE Mini-Project	
Title	College Website
Location	**SRES'COE Kopargaon**
Team Size	4
Technologies	**Database Connectivity, VB**
Role	• As a developer I was responsible to develop a module of College Website. • Study of functional requirements, guidelines to the team members • Coding

Personal Information:	
Name	**PranitShriniwas Thakur.**
Father's Name	ShriniwasKamalakar Thakur.
Date of Birth	2nd January 1986.
Marital Status	Single.
Nationality	Indian.
Permanent Address	2408, Hundiwala Lane, Nr. RanaPratapPutala, Yeola, Dist Nasik, pin 423 401.
Languages	Hindi, English, Marathi, Gujrathi.
Contact Phone	(M) +91-9975772212 (H) 02559-265625.
Email	pranitbhai@rediffmail.com

AUTHOR'S PROFILE :-2. PROF. VIJAY B. PATIL.

First Name: Vijay	Middle Name: Balkrushna	Last Name: Patil	
Position: Post Graduate			
Organization or University: Dr. BabasahebAmbedkarMarathwada University			
Detailed Post Address (Important!): S.T. Colony Plot no. 154, CIDCO N2			
City: Aurangabad	State: Maharashtra	Country:INDIA	Postcode: 431006
Telephone: (0240) 2486017	Fax:	Mobile: 9423395719	Email: vijay.bpatil2006@gmail.com
PAN: AQWPP4511D			

AUTHOR'S PROFILE :- 3. PROF. ABHIJIT S. BODHE.

Name: BodheAbhijitSakharam
[B.E.(Computer),2010]

Mobile No.: *+91-9762657344*
E-mail: *bodhe.abhijit@gmail.com*
: *abodhe@ymail.com*

Educational Qualification:

Degree	College	University	Percentage	Class	Year
M.Tech(CSE)	LKCT,Indore	RGTU,Bhopal	Appear	-	2011
B.E.[Computer]	SRESCOE, Kopargaon	Pune	60.26%	**First Class**	2010
H.S.C.	S.S.G.M. College, Kopargaon.	Pune	61.50%	First Class	**2005**
S.S.C.	K.B.P.Vidyalaya, Kopargaon.	Pune	67.90%	First Class	**2003**

Software Exposures:	
Operating Systems	**MS-DOS,Windows(98/XP/Vista), Linux.**
Languages	*C, C++, Visual Basic, Core JAVA, Basics of HTML*
Database	*Microsoft Access, SQL-Server-2005*
Other Softwares	*Rational Rose, QTP, Selenium, Bugzilla*

PROJECTS

BE Project	
Title	**Credit Card Fraud Detection Using Hidden Markov model (CCFD_HMM)** ***[IEEE-***

	2008 Paper]
Description	**General Overview:** The CCFD_HMM software is basically banking software, which works on Hidden Markov Model (HMM) Algorithm. The aim of the software is to recognise the fraud based on the previous history transaction sequence of the credit card. And also inform the about the fraud via SMS to mobile number of card holder and also to the bank and try to stop the fraud in ATM machines.
Technologies	Java, NetBeans, SQL-server2005
Team Size	5.
Role	• Study of functional requirements and analysis. • As a developer developing some part of Project. • Testing of the designed major modules with integration testing.

Mini-Project In VB.6.0:	
Title	**Hospital Management System.**
Technologies	VB.6.0.
Back End	Microsoft Access.
Team Size	3.
Role	• Study of functional requirements. • Testing of the designed modules.

SEMINAR

Title	**Data Mining-Query Processing.**
Location	SRESCOE Kopargaon.

PROJECT COMPETITION

Title	Credit Card fraud Detection Using HMM.[BE Project]
Location	**PICT,PUNE [INC & IMPETUS-2010]**

Personal Information:	
Name	**AbhijitSakharamBodhe**
Father's Name	SakharamAnantraoBodhe
Date of Birth	1st August 1987
Marital Status	Married
Nationality	Indian
Permanent Address	A/P:"SadguruKrupa Niwas", Sharda Nagar, Yeola Road, Kopargaon Tal: Kopargaon, Dist: Ahmednagar.(M.S) Pin:423601
Languages	Hindi, English, Marathi.