Journal of Advances in Science and Technology

Vol. IV, No. VII, November-2012, ISSN 2230-9659

REVIEW ARTICLE

IMPROVING THE PARTICULAR PROTECTION OF COMPANY WIRELESS/WI-FI NETWORKS USING DAIR

www.ignited.in

Journal of Advances in Science and Technology
Vol. IV, No. VII, November-2012, ISSN 2230-9659

Improving the Particular Protection of Company Wireless/Wi-Fi Networks Using Dair

Nisha Ahuja[1] Dr. Shewata Rani[2]

[1]M. Tech Scholar, Punjab Technical University, Jalandhar, Punjab

[2]Supervisor

----------♦----------

INTRODUCTION

Numerous partnerships make considerable ventures in their wireless base. Case in point, Microsoft's Ieee 802.11 based wireless (Wi-Fi) network comprises of more or less 5,000 right to gain entrance focuses (Aps) supporting 25,000 clients every day in 277 structures, blanket more than 17 million square feet . Notwithstanding the supplies costs, the expenses of arranging, conveying, and keeping up such networks is considerable. Subsequently, it is significant to improve base that enhances the capability of Information Technology (It) divisions to maintain and secure their wireless networks.

Lately, specialists have uncovered security vulnerabilities in Wi-Fi networks . They demonstrated that the Wired Equivalency Methodology (Wep), the prevalent 802.11 security component that generally partnerships were utilizing around then, was essentially imperfect. In an arrangement of remarkably exposed papers, they demonstrated that 802.11 networks could be bargained effectively. The group responded rapidly by improving and conveying substitute security results counting Vpns, Ieee 802.1x , some varieties of Eap , Smart cards, and all the more as of late Wpa . Yet, the wireless Lan (WLAN) security issue was not totally determined.

A year ago, Microsoft led an arrangement of meetings with WLAN executives of numerous vast and minor conglomerations . The objective of these meetings was to grasp the difficulties included in conveying and administering corporate WLANs. The issue of WLAN security came up over and again throughout these meetings. All heads felt that WLAN security was an issue. They were unhappy with the nature of the devices they had at their transfer. A significant number of them would occasionally stroll around their edifices utilizing WLAN checking software (e.g. Netstumbler) searching for security vulnerabilities.

Some contracted exorbitant outside specialists to direct security defenselessness dissections of their WLAN sending, just to infer that what they truly required was an on-going observing also cautioning framework. Generally managers accepted that better frameworks to operate WLAN security are wanted.

Indeed, after methodologies, for example Ieee 802.1x and Wpa are conveyed, corporate networks could be traded off by off-the rack 802.11 fittings and software. Case in point, an unapproved Ap could be joined with the corporate Ethernet, allowing unapproved customers to associate with the corporate network. The rebel Ap may be associated by a noxious individual or, as is more often the case, by an representative who honestly interfaces an Ap in his office without acknowledging that he is bargaining the corporate network. A rebel Ap (or a rebel impromptu network) can dodge the extravagant efforts to establish safety that the It branch might have put set up to ensure the association's learned property.

To test our affirmation that individuals incidentally trade off the security of their networks, we directed an analysis in two substantial conglomerations that had secured their WLANs utilizing one of the systems specified long ago. We strolled around with a WLANenabled smart phone in a little segment of the two grounds searching for Aps to which we could associate. Certainly enough, we discovered numerous "Rogue Aps". We efficaciously joined with the rebel Aps and we were fit to skim the Internet and to enter inward web servers in both conglomerations.

Past rebel Aps and maverick impromptu networks, there are a number of different routes to attack corporate 802.11 networks. For instance, Spying, where the attacker latently listens to the traffic on the wireless network and gathers helpful qualified information, Denial of Service, where an attacker endeavors blemishes in the 802.11 order to handicap the wireless connection and upset conveyance, Phishing (at times called Pharming), where the attacker imitates a honest Ap and draws clueless customers to unite with it. Parts of these and different attacks are given in Section. The focus is that WLAN security presses on to be a test.

The adequacy of any administration result for wireless networks hinges on the capability to perform

www.ignited.in

Rf sensing from a great number of physical areas. This is critical both for scope also for pinpointing the exact area of an issue. We composed the DAIR (Dense Array of Inexpensive Radios) framework for raising wireless network administration provisions that benefit from dense Rf sensing. The temperances of our framework and the distinctive provisions that we plan to assemble are portrayed in our later position paper . In this study, we portray the outline, execution furthermore exhibition of our first wireless administration requisition.

The DAIR methodology is special in that it expands the following two critical perceptions. First, in generally endeavor situations one finds more than enough desktop machines. The machines are ordinarily stationary and are associated with divider power. They have exceptional wired connectivity, save Cpu cycles, free plate space, and rapid Usb ports. Second, inexpensive Usb-based wireless connectors are promptly accessible and their costs press on to fall.1 By appending Usb-based wireless connectors to desktop machines, and devoting the connectors to the errand of following the wireless network, we make an ease following base that is then used to maintain the security of the network.

The first DAIR application that we have built and deployed is an alert system that looks for security breaches in enterprise 802.11 networks. It correctly detects inadvertent security breaches by nonmalicious users, and raises the bar against attacks by malicious users. It does not handle the case where a malicious user employs non-802.11 compliant wireless devices to connect to the network.

We have deployed the DAIR security management application in a *98 m* by *32 m* office building. Our current deployment uses 22 desktop machines. We have written 31,757 lines of C, C++, and C# code to build this system. The average CPU load on each of the desktop machines running the DAIR monitoring software is no more than 2.25%. The average CPU load on the DAIR server varies between 20 to 40% depending on the time of the day. Our server and desktop machines are older models with less CPU horsepower and memory than is typically available in current corporate systems. The additional network traffic due to DAIR is an insignificant 2.5Kbps from each desktop machine.

In summary, the primary contributions of this study are:

- We provide specific examples of why standard authentication and encryption schemes are inadequate to secure corporate Wi-Fi networks, which motivates our solutions based on continuous monitoring of Wi-Fi networks.

- We show that to provide comprehensive coverage for detecting security breaches, a dense deployment of RF sensors is necessary.

- We describe how a scalable system of dense Wi-Fi sensors can be built inexpensively.

- We build such a system and evaluate its performance.

ATTACKS ON WIRELESS/WI-FI NETWORKS

In this segment, we portray a few attacks that network managers of corporate Wi-Fi networks need to make preparations for. We comprehensively order these attacks as detached and animated. The classification is vital for comprehension the qualities and limits of the Dair security administration framework.

Spying : Eavesdropping is a latent attack. The attacker inactively tunes in to the traffic on the wireless network and gathers helpful qualified information. The audience may utilize complex code breaking systems . Countermeasures incorporate utilization of better encryption methods as well as physical efforts to establish safety, for example utilization of radio-misty wallpaper . Detached attacks are difficult, if not unthinkable, to recognize and we don't address them in this study.

Interruption : Any attack that permits a client to addition unapproved access to the network is called an Intrusion attack. Interruption attacks are animated attacks and a few such attacks are conceivable. An attacker can trade off the corporate network by picking up physical access to its wired network and uniting a wireless Ap to it. The Ap makes a "gap" through which unapproved customers can unite, bypassing the extravagant efforts to establish safety that the It section might have put set up. A comparable attack could be conveyed out by utilizing specially appointed wireless networks in place of Aps. A corporate network might additionally be bargained when an attacker finds and utilizes an unsecured Ap associated with the network by a clueless representative. The widespread accessibility of inexpensive, simple todeploy Aps and wireless routers has exacerbated this issue. As specified prior, we discovered numerous unsecured Aps in vast conglomerations.

The Dair security administration framework can discover both rebel Aps and rebel specially appointed networks. A different way a corporate network could be bargained is when an attacker acquires the accreditations (e.g., Wep passwords, Ieee 802.1x certificates) would have done well to associate with the corporate network . The Dair security administration framework can't at present discover such attacks.

Refusal of Service (Dos) : Denial of Service attacks are engaged attacks. A mixture of Dos attacks are conceivable. A few Dos attacks abuse imperfections in the Ieee 802.11 methodology. For instance, a disassociation attack is where the attacker sends an arrangement of fake disassociation or

www.ignited.in

deauthentication messages, initiating real customers to separate from the Ap . In a Nav attack, the attacker produces parcels with substantial span values in the edge header, subsequently driving honest customers to sit tight for long times of time before entering the network . In a Difs attack, the attacker adventures certain timingrelated emphasizes in the Ieee 802.11 methodology to forcefully take bandwidth from genuine clients . In every one of the three cases, the attacker transmits bundles in a strange, either by creating resistant parcels, or by transmitting agreeable bundles at an strangely rate. The Dair security administration framework can identify such attacks. Dos attacks are likewise conceivable by making great measure of Rf commotion in the neighborhood of the network. The Dair security administration framework can identify such attacks by thinking about current perceptions with verifiable information watched from different vantage focuses.

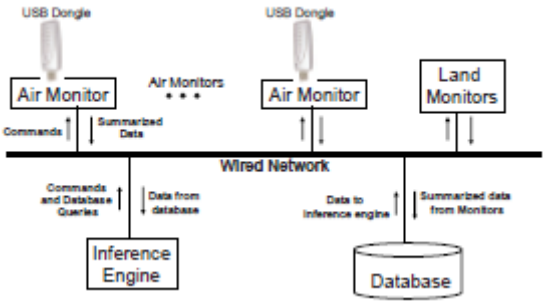

Figure : The DAIR Architecture.

DoS attacks can also be mounted by gaining access to the corporate wired network and attacking the APs from the wired side. The DAIR system does not handle DoS attacks on the wired network.

PHISHING

Phishing is an active attack. An attacker sets up a wireless AP that masquerades as a legitimate corporate AP (same SSID, perhaps even same BSSIDs). If the client does not use mutual authentication, it is possible for the attacker to lure unsuspecting legitimate users to connect to its AP. The attacker can then use a variety of techniques to extract private information (for example, sniff for passwords). The DAIR system can detect phishing attacks. However, we do not describe solutions to phishing attacks in this study.

DESIGN AND ARCHITECTURE

Figure provides a high-level illustration of the major components of the DAIR system. In this section, we provide a detailed description of each of the components and describe the current status of our implementation.

The DAIR system is designed for easy deployment in enterprise environments, both large and small. DAIR makes use of existing enterprise desktop machines for monitoring. In such an environment, the IT department can decide which desktops will be used for monitoring, and they can also manage the process of deploying the DAIR software on such systems. Therefore, we expect that few incentives will be necessary to convince the primary users of these desktop computers to run the DAIR software on their machines.

In a corporate environment, most users do not have administrative privileges to their desktop machines, so they will not be able to tamper with the DAIR software, either purposefully, or inadvertently.

However, we must ensure that the DAIR monitoring software does not adversely impact the interactive performance of desktop computers it runs on.

The AirMonitors : We use the term AirMonitor to refer to an ordinary desktop computer in the enterprise that is equipped with an inexpensive USB 802.11 wireless card and has two components of the DAIR software installed: (1) the AirMonitor service; and (2) a custom device driver that works with any USB wireless card based on the Atheros chipset. The AirMonitor service is user-level code that runs as a Windows service, the equivalent of a daemon on Unix systems.

The primary task of the AirMonitor is to listen continuously, either on a fixed channel, or in scan mode on a sequence of channels. The AirMonitor configures the wireless card in promiscuous mode, so that all 802.11 frames are received, including those destined for other 802.11 stations.

We modified the Windows device driver written by Atheros for their USB 802.11 chipset to support two new capabilities.2 First, we added frame logging support to the driver so that all received 802.11 frames are copied into an in-kernel ring buffer. All frames are copied into this buffer, including those that have decoding errors – only those frames whose preamble cannot be decoded are discarded. Stored along with each frame is additional information about the frame reception, including the signal strength, the channel, and the data rate. We also added support to allow user-level programs to copy the contents of the kernel ring buffer, and to count how many frames are dropped if the ring buffer becomes full.

The other major capability we added to the driver is a new mode that we call “monitor mode.” Monitor mode disables all of the driver’s default scanning behavior. When the driver is not associated with a wireless

network, it performs periodic active and passive scans. An active scan is performed by switching to each channel, issuing a probe request, and then waiting for probe responses from any surrounding access points. Passive scans are done by listening for beacons on each channel, in turn. Monitor mode is useful for two reasons: first, when monitor mode is enabled the AirMonitors become completely passive; second, when a particular channel is selected, the device will not automatically switch to other channels thereby missing some frames on the channel it was supposed to be monitoring.

The AirMonitor service contains all of the user-level code for monitoring. Figure shows a diagram of the AirMonitor service internals. The AirMonitor service enables promiscuous mode, monitor mode, and frame logging at the driver level, at which point all frames are delivered to the service. Within the service, the basic unit of extensibility is a “filter”: each new application built to use the DAIR system installs an application-specific filter that runs inside the AirMonitor service. The Filter Processor takes all frames from the driver and multicasts them to each running filter. The filter’s primary task is to analyze the frames, summarize them in an application-specific manner, and then submit those summaries to the database server. For example, the filter that we use to assist with detecting rogue wireless networks summarizes all SSID’s (network names) and BSSID’s (Access Point MAC addresses) that it overhears,3 and then submits those summaries to the database server every 90 seconds. To ease the task of building a new filter, the AirMonitor service contains a number of support modules.

For example, filters make use of our 802.11 parser module to extract information from the frames, and they make use of our SQL helper module to assist with the task of submitting summaries to the database. The intent is that filters do whatever summarization is sensible to improve the scalability of the system without imposing an undue CPU burden on the AirMonitors – we don’t want to submit every frame that each AirMonitor overhears to the database, yet we also don’t want the AirMonitors to do all of the complex data analysis, which is the responsibility of the inference engine.

The Command Processor module of the AirMonitor service accepts commands from other components of the DAIR system (e.g., the DAIR management console, or one of the inference engines).

Before accepting an incoming request, it checks to see if it can fulfill the request. For example, if an AirMonitor receives a new request to monitor a specific channel different from the one it is already monitoring, it will refuse that new request. Similarly, if the AirMonitor determines that the additional request will place un-due burden on the host, it will refuse the request. While the precise definition of what constitutes undue burden varies based on circumstances, parameters such as history of CPU and memory usage are taken into consideration .

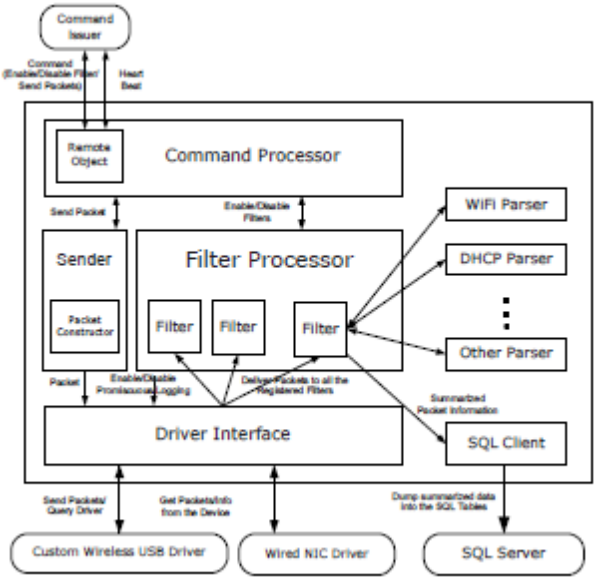

Figure : The AirMonitor Architecture.

The Airmonitor junctions are not restricted to aloof perceptions. Case in point, a deduction motor might solicit one of the Airmonitors to endeavor to connect with an Access Point with a specific end goal to accumulate more informative data. This requires the Airmonitor junction to send cooperation solicits and to process approaching reactions.

The sum of the segments indicated in the Airmonitor administration outline have been enabled. Besides, we as of now have executed four filters: one to condense Ssid and Bssid qualified data for identifying rebel wireless networks, one to skim over disassociation outlines, one to condense information exchanges between customers what's more access focuses, and one to compress outlines that seem to have unusual (Nav) values.

The Landmonitors : Internally, the structure of the Landmonitors is indistinguishable to that of the Airmonitors. The nexus contrast between the Landmonitor also the Airmonitor is that Landmonitors are utilized to screen the wired network – as we will two of our tests for distinguishing maverick wireless networks include screening the wired network.

We need that Landmonitors will be sent with much less thickness than Airmonitors, in spite of the fact that this hinges on upon the network configuration at a given site. For numerous conglomerations, having a solitary Landmonitor for every subnet that consistently screens the uplink to the subnet router will offer sufficient perceivability into the wired network. For those conglomerations that need more amazing perceivability , numerous undertaking class Ethernet switches can alterably empower port

www.ignited.in

reflecting, allowing a Landmonitor to push through numerous distinctive connections within a subnet.

Similarly as with the Airmonitors, all the Landmonitor parts have been actualized, and we have additionally brought about two filters for the Landmonitors: one for screening Dhcp demands, and an alternate one to bring about replay discovery. The parts of these filters are portrayed in the following area.

The Inference Engine : The computationally serious examination undertakings are all performed by the derivation motors. As was the situation with the filters circulating everywhere Screen administration, every provision that is based run on the DAIR framework institutes a provision specific inferencing segment that runs on one of the derivation motor junctions. Our want is that It chairmen will dispense devoted machines to inferencing instead of running these errands on close client's desktop Pcs.

The derivation motors research new occasions by issuing occasional inquiries to the database server. For generally provisions, such questions just need to break down information that was submitted to the database server by the Airmonitors after the past inquiry. To show the kind of processing done by a deduction motor, we briskly depict the derivation motor for discovering rebel wireless networks. The surmising motor issues occasional questions that take a gander whatsoever fresh debuts in the "Ssid and Bssid seen" table since the final question, then after that checks if any of the aforementioned networks are not in the record of sanction Ssid's and Bssid's. In the event that it finds an obscure network, then the surmising motor issues summons to the Airmonitors to perform the succession of tests used to choose whether the obscure wireless network is joined with the corporate wired network being referred to.

The Database : We utilize Microsoft's Sql Server 2005 as our database server. We made no custom modifications to the database server. Besides, separated from making fitting table layouts, records and triggers, we did small to streamline the database exhibition. We want to do further enhancements by utilizing more modern instruments that perform workload-specific file tuning.

The DAIR framework is intended to scale to handle extremely imposing ventures. Our utilization of a concentrated database does not restrain the scale of the framework in light of the fact that when the amount of customers in the framework surpasses the limit of a solitary database server, one can basically send a different database server. The main requirement is that customers ought to be alloted to servers in an area conscious way, to limit the amount of inquiries that must be performed crosswise over various database servers.

DETECTING ATTACKS

We now portray how we power the DAIR structural engineering to recognize interruption and disavowal of administration attacks. Interruption Attacks : We keep tabs on interruption attacks that include association of unapproved wireless gear to a corporate network. There are numerous situations whereby maverick wireless supplies may be joined to a corporate network. Case in point, a representative may acquire a wireless Ap from home and connect it to the corporate network without configuring it to require the indispensible validation.

Additionally a disappointed worker may deliberately append an unapproved Ap to the corporate network. When an unapproved Ap is joined to the corporate network, the security of the network is bargained regardless of the fact that all the sanctioned Aps are configured to use fitting verification instruments. In this manner, locating these unapproved or "maverick" Aps is a vital test.

One may contend that the maverick Ap issue is best settled by securing the wired network. Case in point, if the 802.1x methodology is sent on the wired network, or in the event that some manifestation of Mac address filtering is utilized, unapproved access focuses will not have the ability to interface with the wired network. Correspondingly, Vpn or Ipsec based results can restrain access to corporate assets to approved customers.

While these results are absolutely suitable, they don't completely illuminate the issue. A commissioned customer, joined with the wired network furthermore outfitted with a wireless interface, can connect the two network interfaces to furnish interface layer sending, or furnish Ip-level sending by going about as a Nat. The wireless interface can then be put in specially appointed mode, and used to permit unapproved customers to join to the wired network. For instance, Carnegie Mellon University has as of late issued restrictions against having two engaged interfaces on the same machine. We place that a conveyed overseeing base, acting notwithstanding results like 802.1x and Vpns, furnishes an improved result for the issue.

It might show up at first look that the overseeing foundation does not have to do much: a conglomeration essentially ought to support a database of all commissioned Aps, incorporating their Ssids and Bssids. An alert is raised whenever an obscure Ssid or Bssid is caught by a wireless sensor. This sensor could be an Ap, a versatile customer, or a committed sensor junction. This is the fundamental system proposed in past examination , and numerous wireless administration associations offer rebel Ap location as a major aspect of their item offerings .

Tragically, this straightforward methodology is defenseless to both false negatives and false positives. We now talk over how the DAIR system could be utilized to minimize both false positives and false negatives.

Dos Attacks : Several blemishes in the 802.11 construction modeling could be abused to create a mixture of Dos attacks on corporate Wi-Fi networks. The DAIR system could be utilized to identify a mixed bag of these attacks. We have at present accomplished instruments to catch two sorts of Dos attacks.

Deauthentication / Disassociation Attacks-This attack is depicted in part in Bellardo et al. . The attacker sends spoofed deauthentication or disassociation edges to possibly a versatile customer, a right to gain entrance focus, or both. This will drive the schmuck or victimized individuals to passageway the authenticated/associated state. Since generally wireless drivers on customer apparatuses immediately attempt to re-cohort assuming that disassociation happens, the attacker must enduringly create such spoofed edges to reason significant administration interruption. The attack is especially troubling, since the attacker just should catch outlines from the corporate Wi-Fi network to complete the attack – it doesn't have to do any complex decryption.

We have actualized a filter that permits every Airmonitor to submit disassociation casings to the database. The deduction motor can effortlessly discover the expanded level of disassociation or alternately deauthentication outlines (maybe by relating edges watched at diverse Airmonitors) and raise an alert. Moreover, the surmising motor can additionally furnish a tough situation of the area of the attacker by relating the sign quality of the disassociation/ deauthentication outlines seen by distinctive Airmonitors.

NAV Attacks-In Raya et al. , the creators nitty gritty numerous Dos attacks on Wi-Fi networks. In one of the attacks, the attacker consistently sends outlines with artificially huge span values in the 802.11 header . The span field is utilized to overhaul the network portion vector (NAV) for any unit that catches these edges. Along these lines, these huge NAV qualities will drive the different transmitters in go of the attacker to withhold their transmissions for augmented times of time.

We have actualized a filter that permits the Airmonitors to submit qualified information about casings with unusually length of time qualities to the database. The surmising motor further breaks down these casings to raise a caution if essential. This instrument works by measuring the true length of time of the information transmissions and analyzings them with the term qualities held in the 802.11 header.

This system is basically indistinguishable to that portrayed in Raya et al.. By corresponding perceptions made by distinctive Airmonitors, the surmising motor can additionally give a difficult time of the area of the attacker.

EXPERIMENTAL RESULTS

We built the DAIR system, and it is currently operational in several offices on our floor. We begin by providing a detailed description of the hardware and software that we use. Then, we present results that support our argument that dense deployment is necessary for detecting intrusion attacks. Having argued for dense deployment, we then present results that show that the DAIR architecture is capable of handling the dense deployment. Finally, we present results that illustrate certain aspects of the detection techniques described in Section.

Test Environment : Our experiments were conducted on one floor of a fairly typical office building. Our building has rooms with floor-to-ceiling walls and solid wood doors. There is a corporate wireless LAN with six 802.11 a/b/g access points operating on our floor. Our current DAIR system deployment consists of 22 AirMonitors and 1 database server.

Our database server is Microsoft SQL Server 2005 running on Microsoft Windows Server 2003 SP1. The server hardware is a Compaq Evo D500 with a 1.7 GHz Intel Pentium 4 and 1GB of RAM. The database is stored on a 40 GB 7200 RPM Seagate IDE drive. The drive is formatted as NTFS.

Our AirMonitors run on one of two different types of hardware. One type of machine is a HP Compaq Business Notebook nc6000 with a 2 GHz Intel Pentium M and 512 MB of RAM. These machines run Microsoft Windows XP SP2. We have 6 AirMonitors

of this type, located near the six corporate network access points in our building (offices 17, 26, 27, 28, 29 and 30), and used for the deployment density experiments. The other type is a Compaq Evo D500 SFF, identical to the hardware used for our database server.

These machines run MicrosoftWindows Server 2003 SP1. We have 16 AirMonitors of this type, located in offices numbered 23 and lower, and used for all the other experiments. All AirMonitor machines were equipped with a Netgear WG111U USB 2.0 dongle. This is an 802.11 a/b/g radio with an Atheros chipset.

Sensor Deployment Density : We argue above that the availability of inexpensive, USB-based wireless cards makes dense deployment of wireless sensors possible. However, we have not addressed the question of what deployment density might be sufficient. The results in this section provide some guidance on this point. Given the seriousness of the

rogue AP problem, we would like to maximize the probability of detection, while minimizing the number of false positives and false negatives. Let us consider a scenario in which an unsuspecting user brings an AP or a wireless router from home, and plugs it into the corporate network. The user uses the AP for his/her own work. In this scenario, both the AP and the user's wireless NIC are likely to be operating at full power. In other words, the user is not making any attempts to hide from the monitoring system. We would like to know what density of AirMonitors is required to detect such a rogue device.

System Scalability : We argued that a dense deployment of AirMonitors is necessary for better system performance. However, a dense deployment brings scalability challenges. In this section, we evaluate the scalability of our system.

Each AirMonitor was set to listen on channel 1 in the 802.11b/g band. We selected channel 1 because it is the busiest channel in this corner of the building. We ran the 16 AirMonitors for a period of 24 hours spanning a typical workday.

Each AirMonitor was running 4 filters: the BSSID filter, the Data Packet filter, the Disassociation filter and the NAV filter. Each filter inserted data into the database at regular intervals. The update interval for each filter was chosen at random between 60 and 120 seconds. Thus, the average update interval was 90 seconds. In addition to being an AirMonitor, the machine in office 01 was also acting as a LandMonitor, and ran a DHCP filter.

In addition to the AirMonitors, one instance of the inference engine was also running. The inference engine issued several queries to the database every 60 seconds, checking for rogue networks as well as disassociation and NAV attacks.

During these 24 hours, the average number of packets processed by an AirMonitor was over 4.9 million. The filters summarized the data quite effectively: we estimate that each AirMonitor, on average, generated less than 2.5Kbps of traffic on the wired network.

REFERENCES

- Haddad04 Haddad, Ibrahim and Gordon, David. "The Basics of DNSSEC" ONLamp.com, O'Reilly, October 14, 2006.
- Tzanidakis06 Tzanidakis, Manolis. "Creating a sSecure Linux-based Wireless Access Point" Linux.com. July 19, 2006.
- Egilsson07 Egilsson, Einar. "Redirector :: Firefox Add-ons" Mozilla Software Foundation. October 5, 2007.
- N. Chou, R. Ledesma, Y. Teraguchi, and J. C. Mitchell. Client-side defense against web-based identity theft. In *NDSS*, 2004.
- R. Dhamija and J. D. Tygar. The battle against phishing: Dynamic security skins. In *SOUPS '05: Proceedings of the 2005 symposium on Usable privacy and security*, pages 77– 88, New York, NY, USA, 2005. ACM Press.
- M. Jakobsson. Modeling and preventing phishing attacks. In *Phishing Panel of Financial Crytography*, 2005.
- S. H. Ben Adida and R. Rivest. Fighting phishing attacks: A lightweight trust architecture for detecting spoofed emails. Feb 2005.
- W. A. Arbaugh, N. Shankar, and Y. J.Wan. Your 802.11 wireless network has no clothes. In *IEEEWireless Communications*, 2001.
- P. Bahl and V. N. Padmanabhan. RADAR: An in-building RFbased user location and tracking system. In *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, pages 775–784, 2000.
- M. Bailey, E. Cooke, F. Jahanian, J. Nazario, and D. Watson. The Internet Motion Sensor: A Distributed Blackhole Monitoring System. In *Proceedings of the* 12th *ISOC Symposium on Network and Distributed Systems Security (SNDSS)*, pages 167–179, February 2005.
- R. Beverly and S. Bauer. The spoofer project: Inferring the extent of source address filtering on the internet. In *Proceedings of USENIX Steps to Reducing Unwanted Traffic on the Internet (SRUTI) Workshop*, pages 53–59, July 2005.
- R. A. Beyah, C. L. Corbett, and J. A. Copeland. The case for collaborative distributed wireless intrusion detection systems. In *IEEE International Conference on Granular Computing*, May 2006.
- Bittau, M. Handley, and J. Lackey. The final nail in wep's coffin. In *SP '06: Proceedings of the 2006 IEEE Symposium on Security and Privacy (S&P'06)*, pages 386–400, Washington, DC, USA, 2006. IEEE Computer Society.

- D. P. Blinn, T. Henderson, and D. Kotz. Analysis of a Wi-Fi hotspot network. In *Proceedings of the International Workshop on Wireless Traffic Measurements and Modeling*, June 2005.

- N. Borisov, I. Goldberg, and D.Wagner. Intercepting mobile communications: The insecurity of 802.11. In *Proceedings of ACM Mobicom, Rome, Italy*, July 2001.

- S. Byers, L. F. Cranor, D. P. Kormann, and P. D. McDaniel. Searching for privacy: Design and implementation of a P2Penabled search engine. In D. Martin and A. Serjantov, editors, *Privacy Enhancing Technologies*, volume 3424 of *Lecture Notes in Computer Science*, pages 314–328. Springer, 2004.

- J. Cache and D.Maynor. Device drivers. Presentation at Blackhat USA 2006, August 2006.

- R. G. Cole, N. Phamdo, M. A. Rajab, and A. Terzis. Requirements on worm mitigation technologies in MANETS. In *PADS '05: Proceedings of the 19th Workshop on Principles of Advanced and Distributed Simulation*, pages 207–214, Washington, DC, USA, 2005. IEEE Computer Society.

- G. Portokalidis, A. Slowinska, and H. Bos. Argos: an emulator for fingerprinting zero-day attacks. In *Proc. ACM SIGOPS*

- *EUROSYS'2006*, Leuven, Belgium, April 2006.

- Shannon and D. Moore. The Spread of the Witty Worm. *IEEE Security & Privacy*, 2(4):46–50, July/August 2004.