Wireless Snort – A WIDS in Progess

Wireless Snort – A WIDS in progress

Dr. Craig Valli
School of Computer and Information Science
Edith Cowan University, Western Australia
Email: c.valli@ecu.edu.au

Abstract

The Snort intrusion detection system is a widely used and well-regarded open sourcesystem used for the detection of malicious activity in conventional wired networks. Recently, software patches to enable 802.11 wireless intrusion detection capability in Snort have been released. This paper focuses on the lessons learned from a live deployment of these wireless extensions to the Snort IDS. Generic issues with the deployment of wireless intrusion detection systems are discussed in this paper. In addition, preliminary findings and analysis from the data collected in the pilot study using the wireless enabled snort intrusion detection system are also presented.

Keywords: snort, intrusion detection, wireless, 802.11b, infrastructure

INTRODUCTION

The increasing implementantion of wireless networks and their inherent problems (Valli & Wolski, 2004;Woodward, 2004) has seen the need for reliable intrusion detection systems (IDS) that function in the wireless domain become a necessity. Wireless networks present a special issue for many conventional network intrusion detection systems as wireless networks have relatively insecure Layer 1 and Layer 2 functionality. Most conventional IDSs do not go below the IP layer to detect anomalous behaviours and are largely ineffective at securing the lower layers needed to protect wireless network.

This paper outlines the special problems associated with wireless intrusion detection and a subsequent IDS deployment. The paper then outlines initial findings and observations from preliminary research using a Snort IDS that was patched with wireless extensions.

Snort was primarily chosen because as an IDS it has reached a stage of maturity and development where its functionality and robustness allow for successful deployment into established wired networks. It as a consequence has a wide range of supporting tools and programs for analysis including ACID, snortalog, SnortReport to mention a few. In addition there are numerous books published on the configuration and deployment of the Snort IDS into wired networks.

IS WIRELESS DIFFERENT?

Many would argue that wireless networks are the same as conventional wired networks, and unfortunately make this assumption when producing products for deployment. The simple fact is that wireless networks are not the same as wired networks and have markedly different modes of operation. Subsequently, the resultant threat and intrusion profile whilst including existing wired threats and intrusions now includes a unique set of wireless threats as well.

802.11 wireless is susceptible to standard radio wave transmission jamming which is not in the scope of this paper (Hoad and Jones 2004). However, due to inherent problems within the protocol itself due to its reliance on air as a transmission media, it is susceptible to interception and injection attacks. Unlike conventional networks where the broadcast locus for a network device is the physical confines of the wired infrastructure, the locus for wireless device is not limited to a connected network backplane: it is anything within viable range of the IDS. This presents several problems for IDS deployments.

In a conventional network, IDS are typically placed in an area of the network that has a high degree of homogenity, low levels of inherent entropy and a high level of control on network flows. With wireless networks an IDS may be receiving broadcast from nearby networks in the form of beacon packets, or full data packets that can be anywhere within viable antennae range. This extended, and to a large degree entropic, envelope means that a wireless IDS must process a potentially large number of malicious packets of unknown origin. Conventional IDS and other network countermeasures can perform name resolution and reverse mapping techniques to verify source frames/packets before passing them to the IDS. No such certainty is afforded a wireless IDS.

Wireless signal loss and seepage to other networks is a problem for wireless IDS. Routers, bridges and repeaters on a conventional wired network ensure that the signal that is being transmitted is of sufficient strength to be carried along the network bus. Wireless signal literally leaks into a three dimensional dissipating cloud around the AP focus. This dissipation subsequently leads to the attenuation of the radiated signal, and in particular on the periphery of this cloud leads to corrupted signal and packets. A wireless antennaes *modus operandi* is simply to absorb and process signals that it receives from its environment. This weakness in wireless operation also presents special problems for a wireless IDS in the form of a higher degree of corrupted and nonsensical packets.

Subsequent error rates generated in a wireless network would be considered high error rates for a wired equivalent. This degree of anomaly could cause problems with upper bounds and limits that are programmed as "acceptable" when much of the source code may have been borrowed from, or specifically developed for, wired networks.

The addition of monitoring the lower layers of the network exchange also presents issues that are not the domain of conventional IDS. The extra data gathered adds to complexity of analysis, both at the actual instance of the event, and post event. This data does not eventuate in a conventional ethernet network as Layers 1 and 2 are secure and it is typically not monitored. This is a nexus point in wireless IDS, as to not monitor this activity will allow Layer 2 attacks and vulnerabilities such as disassociation and de-authentication (Abaddon, 2003; Bellardo and Savage, 2003; Valli and Wolski, 2004; Woodward 2004)

Coupled to these problems is the fact that many of the logfile analysis tools do not interact or interrogate layers 1 and 2 data. Analysis console for intrusion data (ACID) is a commonly used tool for analysis of logfiles. Several books have been written that recommend ACID as a suitable analysis tool for use with Snort IDS logfiles, and in fact use this as the central tool (Beale and Foster, 2003;Rehman, 2003).

EXPERIMENTAL SYSTEM

The experimental system is designed as a wireless honeypot and comprises a base install of the Linux Mandrake version 10 Operating system running on a 2.4.25 kernel. The Snort IDS used was version 2.1.1, and was patched to run using the wireless patch provided by Gracia (2004). The patch by Gracia (2004) was an extension of patches provided by Lockhart (2003a). The standard Snort rule sets were modified to reflect a profile used on a existing wired server that exists within the University DMZ. It also had a modified wireless rule set based on the wireless standard rule set (wifi.rules) to suit the configuration of the honeypot server. The wifi.rule rule set was modified such that legitimate APs in the network would not generate Snort Alerts or cause entries in the log files. The patches allow for a variety of wireless IDS capabilities not afforded in the standard Snort distribution that will now be outlined.

Rogue access point (AP) detection allows for the detection of Rogue APs and Ad-Hoc networks that broadcast within the range of the IDS. This works by ignoring service set identifiers (SSID) of the legitimate network, and monitoring all others that appear. A Rogue AP is essentially used as a deceptive mechanism to obtain authentication details from legitimate users of a wireless network. At a more sinister level, with tunneling and cryptographic spoofing techniques used in tools like CrackerJack, the user who believes they are using a legitimate AP is actually having their transmission intercepted. Clients that are also incorrectly configured, and sending probe requests, will connect to a Rogue AP a high risk event. Such an event would allow for viable penetration and attack of that machine and any network that it is authenticated, defeating firewalls and other countermeasures installed at the network layer.

Anti-Stumbler is an extension that detects the presence of people using NetStumbler styles of applications. These programs, like Netstumbler, are created to actively detect APs, and to report information about them. Monitoring for this type of network intelligence gathering by potential attackers works by looking for probe requests that contain NULL SSID entries, which is a fingerprint of such a tool. It should be noted that this extension, whilst detecting active network scanners such as NetStumbler, will not adequately detect passive Kismet style scanners. Passive scanners, like Kismet, achieve network mapping by examining transmitted packets rather than active probing of a network and hence are very difficult to locate.

In addition Wireless Snort adds the following abilities (Table 1) to Snort's conventional capabilities (Lockhart, 2003b).

frame_control tests the entire frame control field	**more_data** tests the more data frame control flag
type tests the 802.11 frame's type	**wep** tests the wep frame control flag
stype tests the 802.11 frame's subtype	**order** tests the order frame control flag
from_ds tests the from distribution system frame control flag	**duration_id** tests the frame's duration/id field
to_ds tests the to distribution system frame control flag	**bssid** tests the frame's BSSID
more_frags tests the more fragments frame control flag	**seqnum** tests the frame's sequence number
retry tests the retry frame control flag	**fragnum** tests the frame's fragment number
pwr_mgmt tests the power management frame control flag	**addr4** tests the frame's 4th address field
	ssid tests the frame's SSID

Table 1 – Snort wireless rule options

It is beyond the scope of this paper to fully explain what each of these particular rule options allow, suffice to say they extend Snort into a potentially competent intrusion detection system for wireless operation.

The system has two wireless network interface cards (WNIC) installed. Both of these are Prism 2.5 chipset based and have standard 2Dbi omni-directional antennas. The first WNIC is setup using HostAP and with a deceptive broadcast SSID of FINANCE to attract potential malicious users. HostAP drivers for Linux, when used in conjunction with a Prism 2.5 card, have the capability to function fully as a wireless access point. The honeyd honeypot is also deployed on this interface as it allows for significant and complete emulation of a network infrastructure. This WNIC is fully logged using the facilities in honeyd and the underlying utilities in the Linux operating system. A raw dump of the network traffic on this interface is written to disk for later forensic analysis. The second WNIC is setup to run as the interface that the wireless patched Snort IDS uses and is running in promiscuous mode to capture all packets. This WINC also uses the HostAP driver set.

PRELIMINARY DATA

The information reported in this paper deals with data collected from the epoch of the WIDS on 2nd May 2004 until 2nd August 2004. Two systems have been used for data collection: The WIDS as described above and a conventional IDS (cIDS) that is located in the University's demilitarised zone (DMZ). The only difference between the systems in terms of IDS and general configuration is that the WIDS is running the wireless extensions to the Snort IDS.

General statistics for the 2 systems are displayed in Table 2.

	Wireless	**Wired**
Total Alerts	590781	10345
Source Addresses	89	441
Destination Addresses	229	126
Unique IP Links	464	8105

Table 2 – General Statistics

The difference in Total Alerts is marked with the WIDS being over 50 times more verbose in level of identified alerts. It should be noted that this is not the total number of security related events, but events that trigger an Alert in the rulesets. Upon investigation of this phenomenon, 86% of these were alerts for Rogue AP beacons. This level of alerts is significant and contributes greatly to information glut and reduced overall system performance. The outcome of this glut is the unnecessary processing of these alerts by the intrusion detection system and any analysis tools used at a later date to interpret the collected data.

It is necessary that beacons be monitored for the detection of rogue access points. The large number of beacons, however, might indicate that there needs to be some serious thinking about how the alerting for this particular wireless hazard is performed. Contributing to this high number of beacon alerts were a sizeable percentage of corrupted wireless packets received by the wireless intrusion detection system. This would further imply the possible need to discount packets that have low signal strength in an attempt to reduce the high number of beacon packets being reported.

Source addresses were swayed in favour of the IDS but this is to be expected due to the embryonic nature of the wireless network development in the University at the time of the analysis period. The destination address statistics are indicative of wireless attackers attempting to penetrate or surveil the University's wireless network.

The Unique IP Links statistics in Table 2 indicate a lower number of attacking addresses. In a wireless network the ratio is much lower, running at about 1:5, but in the wired network this is approximately a 1:20 relationship. This relationship, although seemingly disparate, is not. In a wired network an attack, if a distributed denial of service or as a result of modern viruses or worms, will propagate from a wide range of attacking host IPs. In the case of the wireless network, a distributed denial of service attack is currently physically infeasible due to the low numbers of wireless enabled devices present in the wireless network.

The wireless intrusion detection system detected malicious activity indicating that attackers are utilising advanced attack tools in an attempt to disrupt or penetrate wireless networks. The first detected advanced exploit performed by an intruder was a disassociation and deauthentication attack. The attacker developed a set pattern of attack over a 10-week period, producing the exploit between 11 AM and 1 PM each Tuesday and between 2 PM and 3 PM each Friday. The manner in which the attack was sent would indicate that this was non-target specific and could have used the hunter-killer script facility in the Airjack tool, which produces a similar attack signature. It is very difficult at present to mitigate this form of attack on an 802.11b networking infrastructure. The intrusion detection system recognised the attack but as Wolski & Valli (2004) state this is analogous to detecting nuclear explosions with a geiger counter.

The second form of advanced exploit detected had only been made public 36 hours previous to its first detection on the wireless intrusion detection system. The particular exploit was the CTS exploit, a simple but effective flooding denial of service tool. When broadcast, it makes both APs and clients behave as if the channel were always busy, preventing the transmission of any data over the wireless network (AusCERT, 2004). The use of this particular attack within such a short timeframe might indicate that the attacker was either a highly skilled individual capable of writing that custom automated attack tools, or someone who used a prebuilt tool potentially downloaded from a hacking site.

It should be noted that much of the analysis of these advanced attacks had to be conducted manually. This is due to the current inadequacy of open source analysis tools for use in the 802.11 wireless environment. ACID and other tools do not allow analysis lower than Layer 3 which is where many of the current advanced wireless attacks take place.

CONCLUSION

The development of the wireless intrusion patches for the Snort intrusion detection system, although in their infancy, has uncovered a wide range of issues for further research. All inherent weakness aside, the patched intrusion detection system was capable of detecting advanced wireless exploits that were perpetrated upon the wireless infrastructure.

The inherent verbosity of wireless communications, particularly at Layer 2, causes significant problems for effective alerting and response to wireless network intrusions. This particular issue warrants significant research to overcome these difficulties because much of the current valuable intrusion detection signatures and patterns are lost in a morass of white noise. The adaptation of current investigative tools, so that they have higher degrees of functionality in analysis of wireless intrusion detection data, is another area for significant development and research.

This ongoing research will shortly see the deployment of the wireless intrusion detection and honeypot systems into functioning production 802.11b networks. Data collected from the systems will provide valuable forensic data for detailed analysis of intruder *modus operandi.* The expected outcomes are a better understanding of intruders, and appropriate countermeasures to reduce the threat they represent to modern wireless enabled organisations.

REFERENCES

Abaddon (2003) Airjack, http://802.11ninja.net/airjack/.

AusCERT (2004) AA-2004.02 -- Denial of Service Vulnerability in IEEE 802.11 Wireless Devices, http://www.auscert.org.au/render.html?it=4091, Baird, R. and Lynn, M. (2002) Advanced 802.11b Attack, In Blackhat Briefings 2002, Caesars Palace, Las Vegas, Nevada.

Beale, J. and Foster, J. C. (2003) Snort 2.0 intrusion detection, Syngress, Rockland, USA.

Bellardo, J. and Savage, S. (2003) Disassociation and De-auth attack, In 2003 USENIX Security SymposiumUSENIX.

Gracia, S. (2004) Snort Wireless Patches, http://www.snort-wireless.org

Hoad, R and Jones, A. (2004) Electromagnetic (EM) threats to Information Security - Applicability of the EMC directive and Information Security Guidelines, In 3rd European Conference on Information Warfare, MCIL, University of London, Royal Holloway College, Egham, UK

Lockhart, A. (2003a) Snort Wireless. http://www.snort-wireless.org

Lockhart, A. (2003b) Snort Wireless Users Guide, http://www.snort-wireless.org/docs/usersguide/chap2.html#rule_options,

Osborne, M. (2003) FATAjack, http://www.loud-fat-bloke.co.uk/.

Rehman, R. U. (2003) Intrusion detection with Snort : Advanced IDS techniques using Snort, Apache, MySQL, PHP and ACID, Prentice Hall, Upper Saddle River, New Jersey, USA.

Stock, S. and Beames, K. (2002) FakeAP, Black Alchemy Enterprises.

Valli, C. and Wolski, P. (2004) 802.11b Wireless Networks Insecure at Any Speed, In SAM'04(Eds, Arabnia, H. R., Aissi, S. and Mun, Y.) CSREA Press, Las Vegas, pp. 154-158.

Woodward, A. (2004) In 3rd European Conference on Information Warfare, MCIL, University of London, Royal Holloway College, Egham, UK.

COPYRIGHT