Secure Wireless LANS
With
AWG-1000 / HSG-1000

Whitepapers

March 2005

Version 2.1

© 2005 WiBorne, Inc. All rights reserved

This product or document is protected by copyright and distributed under licenses restricting its use, copying, distribution, decryption, decompilation, and reverse engineering. No part of this product or document may be reproduced in any form by any means without prior written authorization of WiBorne, Inc. or its licensors, if any.

The information in this document is subject to change without notice. This documentation is provided "as is" and all express or implied conditions, representations and warranties, including any implied warranty of merchantability, fitness for a particular purpose or non-infringement, are disclaimed, except to the extent that such disclaims are held to be legally invalid

Blank Page

Functionalities and Design

Summary: This article describes the functionalities and design decisions for our wireless security gateway. In addition, a brief look and explanation of the implementation is also covered.

AWG-1000: Wireless Security Gateway / Controller

HSG-1000: Hotspot Gateway / Controller

Overview

What is the Wireless Security Gateway?

The Wireless Gateway which helps you in building secure mobile infrastructure for your site. This device enables you to quickly set up a network of mobile devices, and provides them access to corporate network resources, e-mail, internet and the intranet while protects your internal intranet from any potential threats from any of these wireless users. It has built-in scalability to meet the demands of growing networks and its plug and play nature provides flexibility for incorporating new nodes quickly and reliably. The advanced administrator interface provides plenty of control for setting up policies and operating in different environments.

The Wireless Security Gateway demonstrates many features offered by our technology including:

- Supports IEEE 802.11-a/b/g.
- Highest level of data encryption using IPSec Tunneling Data Encryption technology.
- Supports wide variety of operating systems, including Microsoft Windows 2000, XP, Windows CE (PocketPC), and Linux operating systems.
- Provides a variety of security functions to ensure strong wireless network security.
- Supports virtually unlimited number of access points to offer extensive coverage and bandwidth.
- Intrusion Detection and Prevention (IDS/IPS) that detects and block illegal attacks.
- Provides flexible and dynamic management for local users.
- Supports seamless roaming from zone to zone when multiple Access Points are connected.

Some benefits from this gateway includes:

- Frees up laptops and computers from network cables.

- Saves costly expenses in network cabling system installation.
- High security for your wireless network, to safeguard proprietary information and network reliability.
- Ultra-reliable, standards-based IEEE 802.11a/b/g wireless LAN networking.
- Free from interference, it coexists with other IEEE 802.11a/b/g Access Points.
- Supports vast majority of notebook PCs equipped with wireless slot.
- Seamless Roaming across different Access Point Zones and persistent session roaming eliminates the need for re-authentication by roaming users.

This white paper discusses some core functions in depth and provides insight from the perspective of the creators.

Wireless LANs and 802.11

Wireless local area networks (WLANs) allow greater flexibility and portability than do traditional wired local area networks (LANs).

Unlike a traditional LAN, which requires a wire to connect a user's computer to the network, a WLAN connects computers and other components to the network using an access point device. An access point communicates with devices equipped with wireless network adaptors; it connects to a wired Ethernet LAN via an RJ-45 port. Access point devices typically have coverage areas of up to 300 feet (approximately 100 meters). This coverage area is called a cell or range. Users move freely within the cell with their laptop or other network device. Access point cells can be linked together to allow users to even "roam" within a building or between buildings.

802.11 refers to a family of specifications developed by the IEEE for wireless LAN technology. 802.11 specifies an over-the-air interface between a wireless client and a base station or between two wireless clients.

There are several specifications in the 802.11 family:

802.11 - applies to wireless LANs and provides 1 or 2 Mbps transmission in the 2.4 GHz band using either frequency hopping spread spectrum (FHSS) or direct sequence spread spectrum (DSSS).

802.11a - an extension to 802.11 that applies to wireless LANs and provides up to 54 Mbps in the 5GHz band. 802.11a uses an orthogonal frequency division multiplexing encoding scheme rather than FHSS or DSSS.

802.11b (also referred to as *802.11 High Rate* or *Wi-Fi*) - an extension to 802.11 that applies to wireless LANs and provides 11 Mbps transmission (with a fallback to 5.5, 2 and 1 Mbps) in the 2.4 GHz band. 802.11b uses only DSSS. 802.11b was a 1999 ratification to the original 802.11 standard, allowing wireless functionality comparable to Ethernet.

802.11g - applies to wireless LANs and provides 20+ Mbps in the 2.4 GHz band.

Security of Wireless LANs

The vulnerability of wireless channel to various attacks makes the wireless LANs inherently insecure. The NIST handbook, An Introduction to Computer Security[1], discusses some of these security threats in wireless LANs. The more immediate concerns for wireless communications are device theft, denial of service, malicious hackers, malicious code, theft of service, and industrial and foreign espionage. Theft is likely to occur with wireless devices because of their portability. Authorized and unauthorized users of the system may commit fraud and theft; however, authorized users are more likely to carry out such acts. Since users of a system may know what resources a system has and the system's security flaws, it is easier for them to commit fraud and theft. Malicious hackers are individuals who break into a system without authorization, usually for personal gain or to do harm. Malicious hackers are generally individuals from outside of an agency or organization (although users within an agency or organization can be a threat as well). Such hackers may gain access to the wireless network access point by eavesdropping on wireless device communications. Malicious code involves viruses, worms, Trojan horses, logic bombs, or other unwanted software that is designed to damage files or bring down a system. Theft of service occurs when an unauthorized user gains access to the network and consumes network resources. Industrial and foreign espionage involves gathering proprietary data from corporations or intelligence information from governments through eavesdropping. In wireless networks, the espionage threat stems from the relative ease with which eavesdropping can occur on radio transmissions.

Attacks resulting from these threats, if successful, place an agency's systems—and, more importantly, its data—at risk. Ensuring confidentiality, integrity, authenticity, and availability are the prime objectives of all government security policies and practices. NIST Special Publication[2] states that information must be protected from unauthorized, unanticipated, or unintentional modification. Security requirements include the following:

- **Confidentiality** – A third party must not be able to eavesdrop on the communication between any two parties.
- **Authenticity -** A third party must be able to verify that the content of a message has not been changed in transit.
- **Non-repudiation -** The origin or the receipt of a specific message must be verifiable by a third party.
- **Accountability -** The actions of an entity must be traceable uniquely to that entity.

---

[1] The NIST Handbook, Special Publication 800-12, *An Introduction to Computer Security.*

[2] NIST Special Publication [2](SP) 800-26, Security Self- Assessment Guide for Information Technology Systems

To date, the list below includes some of the more salient threats and vulnerabilities of wireless systems[3]:

- All the vulnerabilities that exist in a conventional wired network apply to wireless technologies.
- Malicious entities may gain unauthorized access to an agency's computer or voice (IP telephony) network through wireless connections, potentially bypassing any firewall protections.
- Sensitive information that is not encrypted (or that is encrypted with poor cryptographic techniques) and that is transmitted between two wireless devices may be intercepted and disclosed.
- Denial of service (DoS) attacks may be directed at wireless connections or devices.
- Malicious entities may steal the identity of legitimate users and masquerade as them on internal or external corporate networks.
- Sensitive data may be corrupted during improper synchronization.
- Malicious entities may be able to violate the privacy of legitimate users and be able to track their physical movements.
- Malicious entities may deploy unauthorized equipment (e.g., client devices and access points) to surreptitiously gain access to sensitive information.
- Handheld devices are easily stolen and can reveal sensitive information.
- Data may be extracted without detection from improperly configured devices.
- Viruses or other malicious code may corrupt data on a wireless device and be subsequently introduced to a wired network connection.
- Malicious entities may, through wireless connections, connect to other agencies for the purposes of launching attacks and concealing their activity.
- Interlopers, from inside or out, may be able to gain connectivity to network management controls and thereby disable or disrupt operations.
- Malicious entities may use a third party, untrusted wireless network services to gain access to an agency's network resources.
- Internal attacks may be possible via ad hoc transmissions.

Solutions to Secure Wireless LANs

Wired Equivalent Privacy (WEP)

The 802.11 standard for wireless LAN communications introduced the Wired Equivalent Privacy (WEP) protocol in an attempt to address the problems discussed above and bring the security level of wireless systems closer to that of wired ones. The primary goal of WEP is to protect the confidentiality of user data from eavesdropping. WEP is part of an

[3] For more detailed information on the risk mitigation and safeguard selection process, refer to NIST SP 800-12, *An Introduction to Computer Security,* and 800-30*, Risk Management Guide for IT Systems*

international standard; it has been integrated by manufacturers into their 802.11 hardware and is currently in widespread use.

If a user activates WEP, the NIC encrypts the payload (frame body and CRC) of each 802.11 frame before transmission using an RC4 stream cipher provided by RSA Security. The receiving station, such as an access point or another radio NIC, performs decryption upon arrival of the frame. As a result, 802.11 WEP only encrypts data between 802.11 stations. Once the frame enters the wired side of the network, such as between access points, WEP no longer applies.

Unfortunately, WEP falls short of accomplishing its security goals as pointed out by Borisov et al.[4] Despite employing the well-known and believed-secure RC4 cipher*, WEP contains several major security flaws.* The flaws give rise to a number of attacks, both passive and active, that allow eavesdropping on, and tampering with, wireless transmissions.

WiFi Protected Access (WPA)

WiFi[5] Protected Access (WPA), which is being promoted by the WiFi Alliance, is an interim security solution that does not require a hardware upgrade in existing 802.11 equipment. WPA is not a perfect solution but is an attempt to quickly and proactively deliver enhanced protection–to address some of the problems with WEP. WPA includes two main features:

1. 802.1X: Framework for Authentication

Combined with an authentication protocol, such as EAP-TLS, LEAP, or EAP-TTLS, IEEE 802.1X provides port-based access control and mutual authentication between clients and access points via an authentication server. The use of digital certificates makes this process very effective. 802.1X also provides a method for distributing encryption keys dynamically to wireless LAN devices, which solves the key reuse problem found in the current version of 802.11.

Microsoft supports 802.1X in Windows XP, and many vendors offer 802.1X in wireless LAN devices, 802.11i is including 802.1X in 802.11 standards.

---

[4] Intercepting mobile communications: the insecurity of 802.11, Proceedings of the 7th annual international conference on Mobile computing and networking Rome, Italy, Pages: 180 – 189, 2001

[5] WiFi means “wireless fidelity” and is a synonym for 802.11b.

2. TKIP: Interim Encryption Solution

The temporal key integrity protocol (TKIP), initially referred to as WEP2, is an interim solution that fixes the key reuse problem of WEP, that is, periodically using the same key to encrypt data. The TKIP process begins with a 128-bit "temporal key" shared among clients and access points. TKIP combines the temporal key with the client's MAC address and then adds a relatively large 16-octet initialization vector to produce the key that will encrypt the data. This procedure ensures that each station uses different key streams to encrypt the data.

TKIP uses RC4 to perform the encryption, which is the same as WEP. A major difference from WEP, however, is that TKIP changes temporal keys every 10,000 packets. This provides a dynamic distribution method that significantly enhances the security of the network.

An advantage of using TKIP is that companies having existing WEP-based access points and radio NICs can upgrade to TKIP through relatively simple firmware patches. In addition, WEP-only equipment will still interoperate with TKIP-enabled devices using WEP. TKIP is a temporary solution, and most experts believe that stronger encryption is still needed.

Virtual Private Networks (VPNs) and IPSec

VPN technology is a rapidly growing technology that provides secure data transmission across public network infrastructures. VPNs have in recent years allowed corporations to harness the power of the Internet for remote access. Today, VPNs are typically used in three different scenarios: for remote user access, for LAN-to-LAN (site-to-site) connectivity, and for extranets. VPNs employ cryptographic techniques to protect IP information as it passes from one network to the next or from one location to the next. Data that is inside the VPN “tunnel”—the encapsulation of one protocol packet inside another—is encrypted and isolated from other network traffic.

IPSec stands for **I**nternet **P**rotocol **SEC**urity. It is a set of protocols developed by the IETF to support secure exchange of packets at the IP layer. IPSec has been deployed widely to implement VPNs.

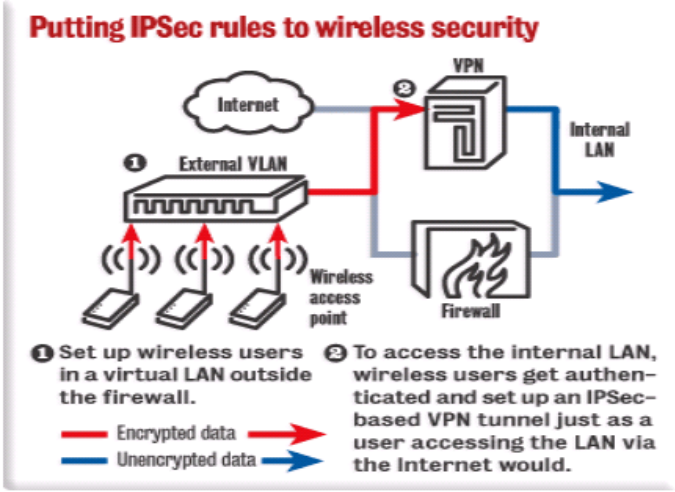
IPSec uses strong cryptography to provide both authentication and encryption services. Authentication ensures that packets are from the right sender and have not been altered in transit. Encryption prevents unauthorized reading of packet contents. Cryptographic algorithms that asyndeta™ applied, are triple-DES, Blowfish, and MD5. It provides up-to 168-bits high encryption.

IPSec supports two encryption modes: *Transport* and *Tunnel*. Transport mode encrypts only the data portion (payload) of each packet, but leaves the header untouched. The more secure Tunnel mode encrypts both the header and the payload. On the receiving side, an IPSec-compliant device decrypts each packet.

For IPSec to work, the sending and receiving devices must share a public key. This is accomplished through a protocol known as Internet Security Association and Key Management Protocol/Oakley (ISAKMP/Oakley), which allows the receiver to obtain a public key and authenticate the sender using digital certificates.

These services allow building secure tunnels through un-trusted networks. Everything passing through the un-trusted net is encrypted by the IPSEC gateway machine and

decrypted by the gateway at the other end. The result is **V**irtual **P**rivate **N**etwork or (VPN). This is a network which is effectively private even though it includes machines at several different sites connected by the insecure Internet.



ISAKMPD and IKE

In IPSec, the client and the server communicate using ISAKMPD. ISAKMP defines procedures and packet formats to establish, negotiate, modify and delete Security Associations (SA). SAs contain all the information required for execution of various network security services, such as the IP layer services (such as header authentication and payload encapsulation), transport or application layer services, or self-protection of negotiation traffic. ISAKMP defines payloads for exchanging key generation and authentication data. These formats provide a consistent framework for transferring key and authentication data which is independent of the key generation technique, encryption algorithm and authentication mechanism.

ISAKMP is distinct from key exchange protocols in order to cleanly separate the details of security association management (and key management) from the details of key exchange. There may be many different key exchange protocols, each with different security properties. However, a common framework is required for agreeing to the format of SA attributes, and for negotiating, modifying, and deleting SAs. ISAKMP serves as this common framework.

The client communicates with the ISAKMPD gateway using the internet key exchange (IKE) protocol. IKE automatically negotiates IPSec SAs and enables IPSec secure communications without costly manual pre-configuration.

Specifically, IKE provides these benefits:

- Eliminates the need to manually specify all the IPSec security parameters in the crypto maps at both peers.

- Allows you to specify a lifetime for the IPSec security association.
- Allows encryption keys to change during IPSec sessions.
- Allows IPSec to provide anti-replay services.
- Permits Certification Authority (CA) support for a manageable, scalable IPSec implementation.
- Allows dynamic authentication of peers.

Automated Pushing for IPSec Client Software

Client authentication for AWG-1000 is designed to be user friendly so that the user doesn't need to install client software separately. All utilities are pushed to client's machine during initial authentication. To ensure maximum security, the client certificate bundle (the client certificate, client key and the CA certificate) which is used to establish the VPN/IPSec tunnel is created during each login session. System then configures VPN automatically for you.

It also supports watchdog to monitor IPSec connectivity between clients and gateway. For each user login, AWG-1000 issues new session x509 key and push it to client machine, it then start negotiating session key to construct tunnel.

There has a system tray that indicates signal strength, or packet service between client's machine and AWG-1000.

Authentication Protocols for WLAN

Modern computer systems provide service to multiple users and require the ability to accurately identify the user making a request. In traditional systems, the user's identity is verified by checking a password typed during login; the system records the identity and uses it to determine what operations may be performed. The process of verifying the user's identity is called *authentication*. Password based authentication is not suitable for use on computer networks. Passwords sent across the network can be intercepted and subsequently used by eavesdroppers to impersonate the user. Especially in WLANs, strong authentication mechanisms are required. Here we discuss some such protocols based on cryptography, which do not allow an attacker to gain any information that would enable it to falsely claim another's identity

Kerberos Authentication

Kerberos is a distributed authentication service that allows a process (a client) running on behalf of a principal (a user) to prove its identity to a verifier (an application server, or just server) without sending data across the network that might allow an attacker or

the verifier to subsequently impersonate the principal. Kerberos optionally provides integrity and confidentiality for data sent between the client and server. Kerberos was developed in the mid-'80s as part of MIT's Project Athena [6]. There are two widely used versions of Kerberos, V4 and V5.

The Kerberos Authentication System uses a series of encrypted messages to prove to a verifier that a client is running on behalf of a particular user. The Kerberos protocol is based in part on the Needham and Schroeder authentication protocol[7], but with changes to support the needs of the environment for which it was developed. Among these changes are the use of timestamps to reduce the number of messages needed for basic authentication, the addition of a ``ticket-granting'' service to support subsequent authentication without re-entry of a principal's password, and different approach to cross-realm authentication (authentication of a principal registered with a different authentication server than the verifier).

RADIUS Authentication

The Remote Authentication Dial In User Service (RADIUS) protocol (RFC 2865) was originally defined to enable centralized authentication, authorization, and access control (AAA) for SLIP and PPP dial-up sessions -- like those made to a dial-up ISP. Instead of requiring every Network Access Server (NAS) to maintain a list of authorized usernames and passwords, RADIUS Access-Requests were forwarded to an Authentication Server, commonly referred to as an AAA Server. This architecture made it possible to create a central user database, consolidating decision-making at a single point, while allowing calls to be supported by a large, physically distributed set of NASs.

When a user connects, the NAS sends a RADIUS Access-Request message to the AAA Server, relaying information like the user's name and password, type of connection (port), NAS identity, and a message Authenticator.

Upon receipt, the AAA Server uses the packet source, NAS identity, and Authenticator to determine whether the NAS is permitted to send requests. If so, the AAA Server tries to find the user's name in its database. It then applies the password and perhaps other attributes carried in the Access-Request to decide whether access should be granted to this user.

Depending upon the authentication method being used, the AAA Server may return a RADIUS Access-Challenge message that carries a random number. The NAS relays the challenge to the remote user (for example, using CHAP). The user must respond with the correct value to prove its asserted identity (for example, encrypting the challenge with its password), which the NAS relays to the AAA Server inside another RADIUS Access-Request message.

---

[6] G. A. Champine, D. E. Geer, Jr., and W. N. Ruh. Project Athena as a distributed computer system. *IEEE Computer*, 23(9):40-51, September 1990.

[7] R. M. Needham and M. D. Schroeder. Using encryption for authentication in large networks of computers. *Communication of the ACM*, 21(12):993-999, December 1978.

If the AAA Server is satisfied that the user is authentic and authorized to use the requested service, it returns a RADIUS Access-Accept message. If not, the AAA Server returns a RADIUS Access-Reject message and the NAS disconnects the user.

When an Access-Accept message is received and RADIUS Accounting is enabled, the NAS sends a RADIUS Accounting-Request (Start) message to the AAA Server. The Server adds an accounting record to its log and acknowledges the request, whereupon the NAS activates the user's session. At the end of the session, a similar RADIUS Accounting-Request (Stop) message is exchanged so that accounting records will reflect the actual session duration and disconnect reason.

In a wireless network that uses 802.1X Port Access Control, the wireless station plays the role of the Remote User and the wireless AP plays the role of the Network Access Server. Instead of connecting to the NAS with a dial-up protocol like PPP, wireless stations associate to the AP using 802.11 protocols.

Once associated, the wireless station sends an EAP-Start message to the AP. The AP requests the station's identity and relays it to an AAA Server inside the RADIUS Access-Request User-Name attribute.

The AAA Server and wireless station complete the authentication process by relaying RADIUS Access-Challenge and Access-Request messages through the AP. Depending upon the EAP type, messages may be carried inside an encrypted TLS tunnel.

If the AAA Server issues an Access-Accept message, the AP and wireless station complete a handshake to generate session keys used by WEP or TKIP to encrypt data. At that point, the AP unblocks the port and the wireless station can send data and receive data to and from the attached network.

If the AAA Server issues an Access-Reject message, the AP disassociates the station. The failed station can try to authenticate again, but the AP prevents the station from actually sending data through the AP into the adjacent network. Note that the failed station can still listen to data sent by other stations -- that is the nature of a radio network, and why it's important to encrypt data sent over the air.

The Attribute-Value pairs included in RADIUS messages can be used by the AAA Server to deliver session parameters to the AP and wireless station, like Session-Timeout or VLAN tag (Tunnel-Type=VLAN, Tunnel-Private-Group-ID=tag). Exactly what additional information can be delivered and used depends on the type of AAA Server, AP, and station products.

LDAP Authentication

LDAP refers to Lightweight Directory Access Protocol[8]. It is a standard protocol based on X.500 that can be used to access information over a network. LDAP is based on entries which are collection of attributes and/or globally unique distinguished name (DN) – like

[8] Described in RFC1777 and RFC2251

domain name. These entries are arranged in hierarchical tree structure. Using LDAP, it is possible to centrally manage users, groups and other data. It does not need to manage separate directories for each application - avoids the “N + 1 directory problem”. It provides distributed management of data to appropriate people and allows users to find data they need. Date is not locked into a particular server. One very important application of LDAP is for authentication and authorization. The client first authenticates to directory, opens TCP connection, provides distinguished name and credentials, server checks whether credentials are correct and accordingly returns the result to the client. If credentials fail, the server returns an anonymous bind. The authentication lasts while connection is open, or until client re-authenticates. Credentials can be password, digital certificates etc.

There are 3 basic levels of authentication in LDAP. First, anonymous: which requires no username and password, and is enabled by default. Second, unauthenticated : which requires only username (sometimes called reference bind). And the third, authenticated, which requires both username and correct password.

Firewall and Walled-Garden

A firewall is a system or group of systems that enforces an access control policy between two networks. The actual means by which this is accomplished varies widely, but in principle, the firewall can be thought of as a pair of mechanisms: one which exists to block traffic, and the other which exists to permit traffic. Some firewalls place a greater emphasis on blocking traffic, while others emphasize permitting traffic. Probably the most important thing to recognize about a firewall is that it implements an access control policy. If you don't have a good idea of what kind of access you want to allow or to deny, a firewall really won't help you. It's also important to recognize that the firewall's configuration, because it is a mechanism for enforcing policy, imposes its policy on everything behind it. Administrators for firewalls managing the connectivity for a large number of hosts therefore have a heavy responsibility.

On the Internet, a walled garden refers to a browsing environment that controls the information and Web sites the user is able to access. This is a popular method used by ISPs in order to keep the user navigating only specific areas of the Web, whether for the purpose of shielding users from information -- such as restricting children's access to pornography -- or directing users to paid content that the ISP supports. America Online is a good example of an ISP that places users in a walled garden.

The term *walled garden* clearly applies in the scenario where wireless LANs and wired LANs coexist. In such a case, the users of clients that are allowed to access the wireless side of the network might not be allowed to access the wired side and vice versa.

AWG-1000: A Secure Wireless Gateway

The WiBorne, Inc., has developed a secure platform for integrated wireless networks. The platform, called AWG-1000, is based on a 1u rackmount appliance running a hardened, ultra-secure operating system. AWG-1000 secures the wireless LANs based on the mechanisms discussed above. Key features of AWG-1000 include:

- Secures 802.11 WLANs (a, b, g); immediate cross-vender support for an expandable number of external 802.11 APs
- Clients supported: Windows 2000, XP, PocketPC, MacOS, and most Unix variants.
- IPSec and SSL/TLS for strong client-to-gateway VPN and VLAN Security
- Seamless IP roaming
- Secure single sign-on integrated with local and domain authentications (Kerberos V, RADIUS, LDAP).
- Quality of Service (QoS) and Bandwidth Management functions
- Comprehensive stateful packet filter (Firewall)
- 802.1X support with EAP, TLP, TLS, MD5, and PEAP for port authentication.
- Proactive host security measures
- Guest/role accounts, with option to bypass VPN.
- Detailed billing system and logon redirecting.
- Ultra-secure OpenBSD Operating System.
- Onboard 802.11b AP. Support for 802.11a/b/g
- WLAN DHCP server, NAT gateway, DNS server
- Configuration backup
- Centralized AP logging
- Consolidated output to remote syslog server
- Load balancing
- SSL/TLS web administration, with real-time traffic monitoring and analysis system
- Serial console administration
- Dynamic and customizable guest accounts, with the intranet fully protected.
- Client software is automatically pushed with initial web-based login, no extra client licenses.
- Hardware cryptographic acceleration to handle high traffic volume
- Standard configuration provides up to 6 10/100Mbps Ethernet ports for flexible connectivity options

Intrusion Detection / Prevention System

An intrusion occurs when a hacker or cracker attempts to break into or misuse your system (for example, stealing the password, altering the email system for spam, deleting crucial system files, etc.). An ***I****ntrusion* ***D****etection* ***S****ystem* (IDS) is a system for detecting such intrusions. IDS can be broken down into several categories according to the source of intrusion. Among them, ***N****etwork* ***I****ntrusion* ***D****etection* ***S****ystem* (NIDS) becomes the most commonly-used one as more and more network activities are involved into people's everyday life.

NIDS has become an integral part of any network security architecture. It monitors packets on the network, wired or wireless, and attempts to discover if a hacker/cracker is attempting to break into a system or cause a denial of service attack. They provide a layer of defense which monitors for predefined suspicious activities or patterns, and alert system administrators when potential hostile traffic is detected.

After detecting such potential attacks and alert administrators, the next step is to protect your network and prevent potential damages. ***I**ntrusion **P**revention **S**ystem* (IPS) does this. Systems with IPS enabled can automatically react to such hostile traffic by blocking them, logging the evidence, emailing the administrator, and so on.versa.

1. Intrusion Detection System (IDS)

Our NIDS is a lightweight, yet powerful network intrusion detection tool capable to detect a wide variety of suspicious network traffic as well as outright attacks. It can provide administrators with enough data to make informed decisions on the proper course of action in the face of suspicious activity.

a. Features

Differing from other packet sniffers such as tcpdump, our NIDS inspects not only the packet header, but also the packet payload. It decodes the application layer of a packet and can be given rules to collect traffic that has specific data contained within its application layer. This allows it to detect many types of hostile activity, including buffer overflows, CGI scans, or any other data in the packet payload that can be characterized in a unique detection fingerprint.

Our NIDS focuses on collecting packets as quickly as possible and processing them in the detection engine. Moreover, it can be configured and left running for long periods of time without requiring monitoring or administrative maintenance.

b. How does it work?

Our NIDS' architecture is focused on performance, simplicity, and flexibility. There are three primary subsystems that make up our NIDS: the packet decoder, the detection engine, and the logging and alerting subsystem. These subsystems ride on top of the libpcap promiscuous packet sniffing library, which provides a portable packet sniffing and filtering capability. Program configuration, rules parsing, and data structure generation takes place before the sniffer section is initialized, keeping the amount of per packet processing to the minimum required to achieve the base program functionality.

***The packet decoder*:** the decode engine is organized around the layers of the protocol stack present in the supported data-link and TCP/IP protocol definitions. Each subroutine in the decoder imposes order on the packet data by overlaying data structures on the raw network traffic. These decoding routines are called in order through the protocol stack, from the data link layer up through the transport layer, finally ending at the application layer. Speed is emphasized in this section, and the majority of the functionality of the decoder consists of setting pointers into the packet data for later analysis by the detection engine. Our NIDS provides decoding capabilities for Ethernet, SLIP, and raw (PPP) data-link protocols.

***The detection engine*:** our NIDS maintains its detection rules in a two dimensional linked list of what are termed Chain Headers and Chain Options. These are lists of rules that have been condensed down to a list of common attributes in the Chain Headers, with the detection modifier options contained in the Chain Options. For example, if forth five CGI-BIN probe detection rules are specified in a given detection library file, they generally all share common source and destination IP addresses and ports. To speed the detection processing, these commonalities are condensed into a single Chain Header and then individual detection signatures are kept in Chain Option structures. These rule chains are searched recursively for each packet in both directions. The detection engine checks only those chain options which have been set by the rules parser at run-time. The first rule that matches a decoded packet in the detection engine triggers the action specified in the rule definition and returns.

The logging/alerting subsystem: packets are logged into an alert file in real-time. It writes a condensed subset of the header information to the alert file, allowing greater performance under load. In order for administrators to view the traffic better, we process the alert file, collect the statistics, and generate the corresponding reports. The reports contain the overall statistics, tables, and diagrams to help better visualize the traffic. Real-time web interface is provided, which can be refreshed automatically.

c. Rules

The rules are simple to write, yet powerful enough to detect a wide variety of hostile or merely suspicious network traffic. There are three base action directives that can be used when a packet matches a specified rule pattern: pass, log, or alert. Pass rules simply drop the packet; log rules write the full packet to the logging routine that was user selected; alert rules generate an event notification and write them into the alert file to enable later analysis.

The most basic rules contain only protocol, direction, and the port of interest. The following is an example:

```
log tcp any any → 10.1.1.0/24 79
```

This rule would record all traffic inbound for port 79 (finger) going to the 10.1.1 class C network address space. Keywords are enclosed in parentheses as “option fields”. Option fields are available for all rule types and may be used to generate complex behaviors from the program. The following example illustrates it:

```
alert tcp any any -> 10.1.1.0/24 80 (content: "/cgi-bin/phf";

msg: "PHF probe!";)
```

The above rule would detect attempts to access the PHF services on any of the local network’s web servers. If such a packet is detected on the network, an event

notification alert is generated and then the entire packet is logged via the logging mechanism.

d. Web Interface

Our NIDS provides simple interfaces for web-accesses. The following is a snapshot taken from our NIDS:

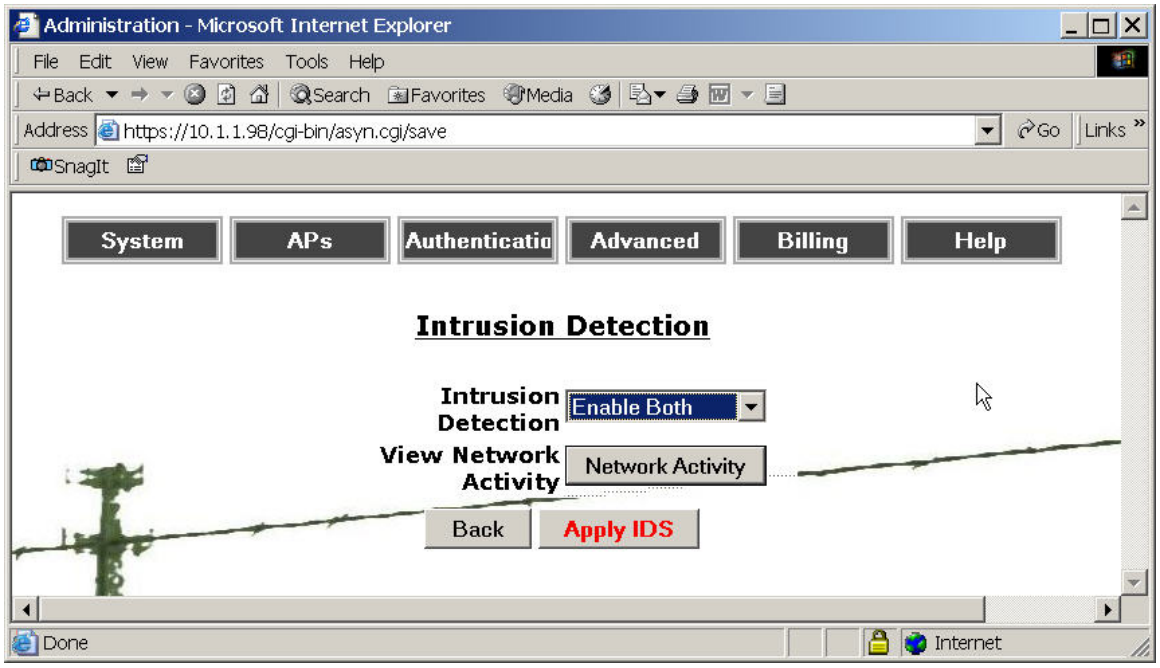


Figure 1: Intrusion Detection Web Interface

You can view network statistics generated from the alert file by clicking on "Network Activity". Suspicious events and malicious hosts are categorized and displayed. Graphs and tables are provided according to their distribution for better viewing. Two types of statistics are collected and displayed:

1) *General Statistics:*

- The distribution of event by hour
- The distribution of event by day
- Popularity of one source host
- Popularity of one destination host
- The distribution of event by destination port
- The distribution of event by protocols
- The distribution of event type of log

2) *Specific Statistics:*

- Events from one host to any with same method
- Events to one host from any with same method
- Events from a host to a destination
- Events to one destination port grouped by attach

- Distribution of attack methods
- Distribution of classification method
- The distribution of event by severity
- Events by hour

The default NIDS configuration enables about 1300 rules. To reduce the amount of alerts and avoid false alarms, some entire rule types such as policy, chat, and virus, are disabled by default, and can be customized and deployed in the future.

The following snapshot shows the general information of IDS statistics such as the time this report was generated, log begin time, end time, length of logs, percentage of logs dropped, total event, source IP recorded, and Destination IP recorded. You can click on the links to link to the corresponding diagrams.

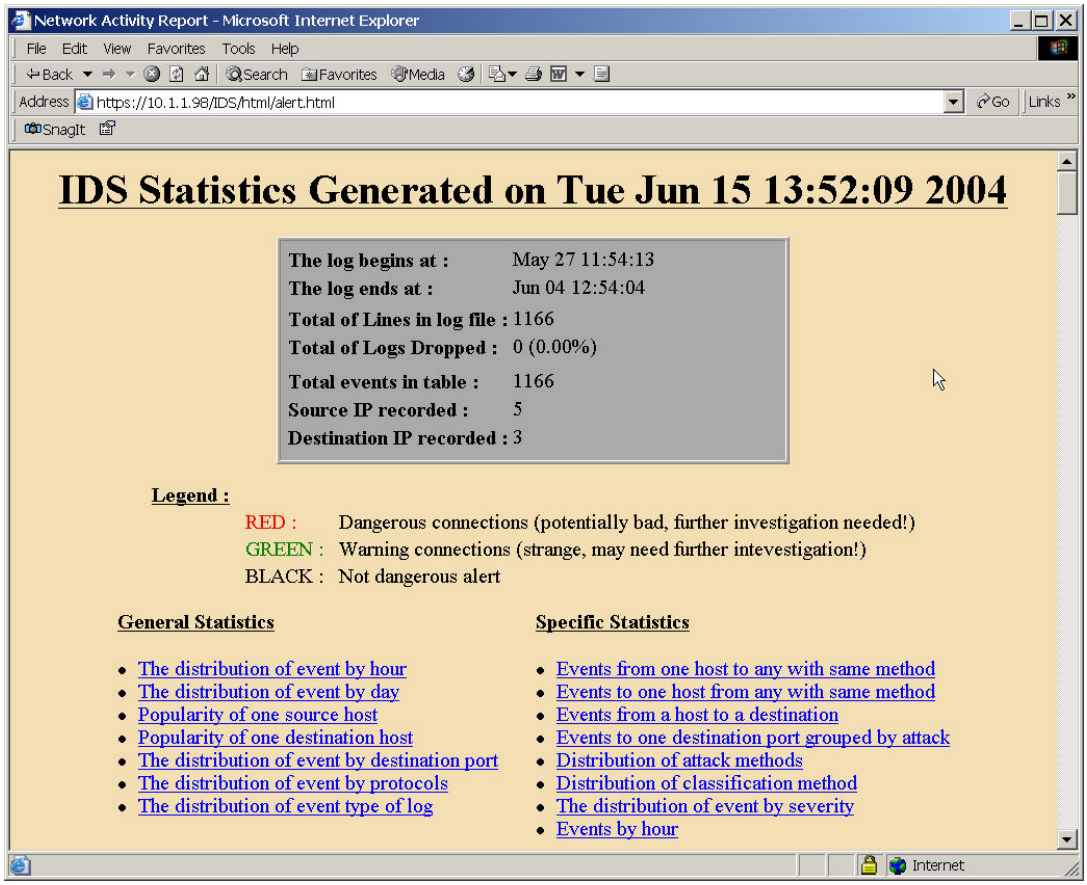


Figure 2: Intrusion Detection Network Activity – General Statistics

The following snapshot shows the distribution of event by protocols and the distribution of severity in pie diagrams. The tables on the left hand side summarize the statistics:

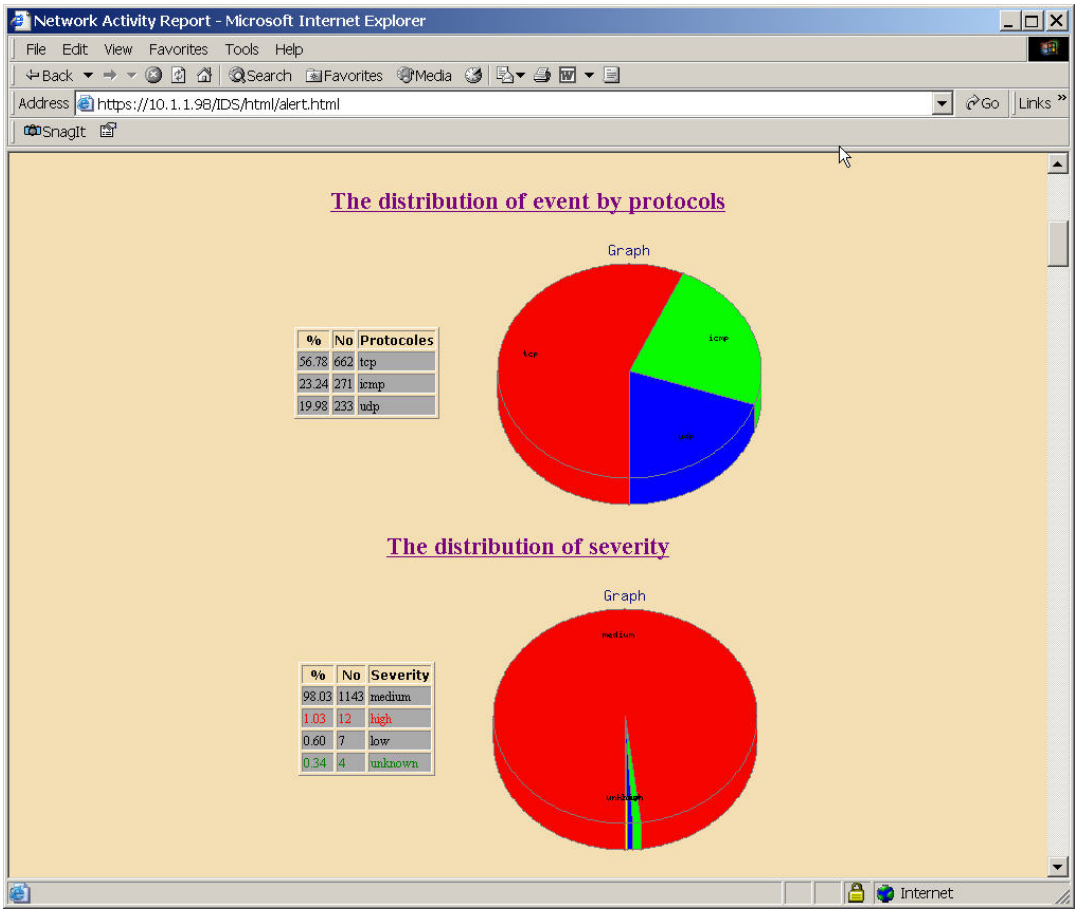
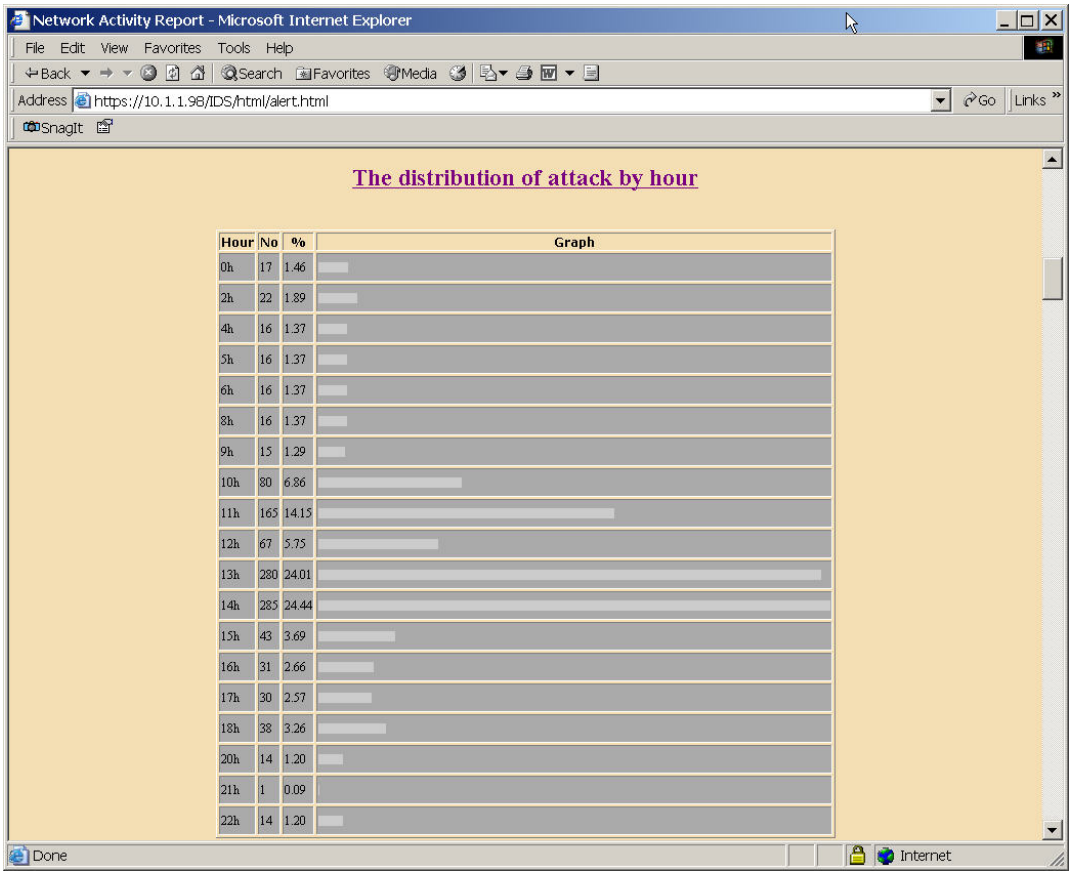


Figure 3: Intrusion Detection Network Activity – Event Distribution

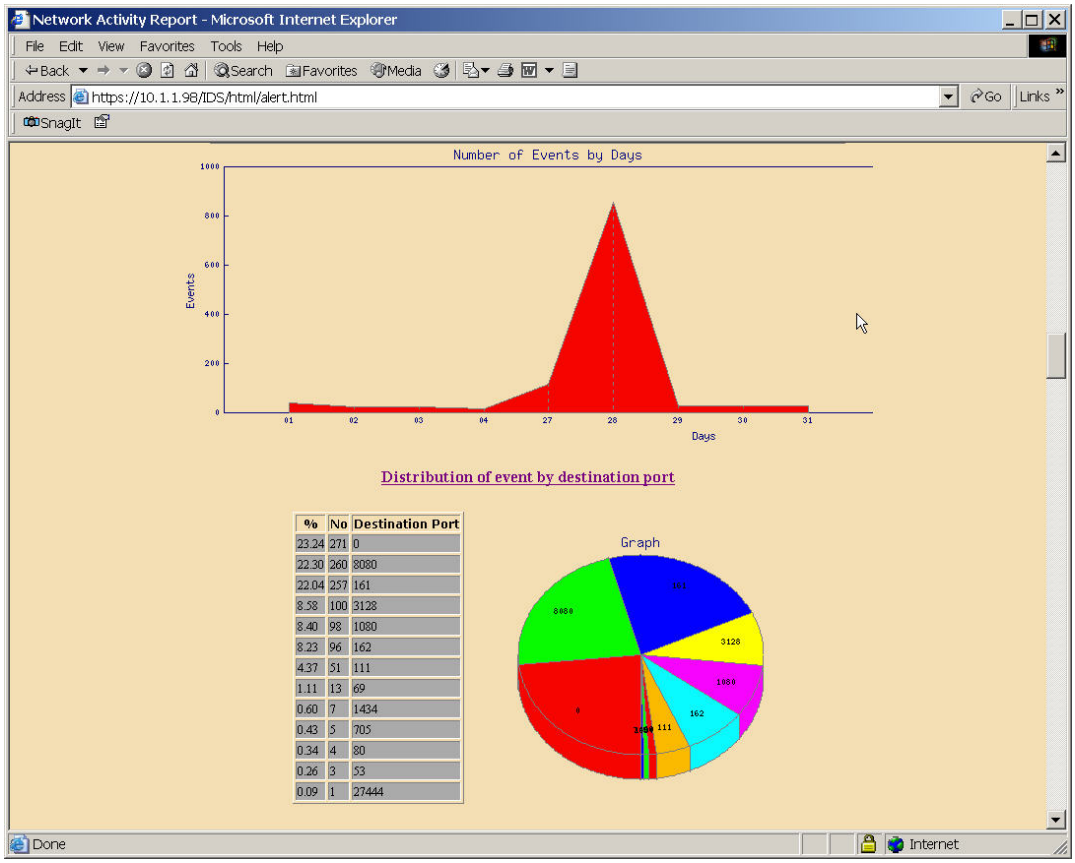
The following table shows the distribution of attacks by hour. The attacks during the past 24 hours are collected and displayed in chronological order.



Hour	No	%	Graph
0h	17	1.46	
2h	22	1.89	
4h	16	1.37	
5h	16	1.37	
6h	16	1.37	
8h	16	1.37	
9h	15	1.29	
10h	80	6.86	
11h	165	14.15	
12h	67	5.75	
13h	280	24.01	
14h	285	24.44	
15h	43	3.69	
16h	31	2.66	
17h	30	2.57	
18h	38	3.26	
20h	14	1.20	
21h	1	0.09	
22h	14	1.20	

Figure 4: Intrusion Detection Network Activity – Attack Distribution by Hour

The following diagrams show the number of events by days and distribution of event by destination port.



%	No	Destination Port
23.24	271	0
22.30	260	8080
22.04	257	161
8.58	100	3128
8.40	98	1080
8.23	96	162
4.37	51	111
1.11	13	69
0.60	7	1434
0.43	5	705
0.34	4	80
0.26	3	53
0.09	1	27444

Figure 5: Intrusion Detection Network Activity – Attack Distribution by Port

2. Intrusion Prevention System (IPS)

Our NIPS is implemented by running a small Perl daemon which parses the NIDS' alert file and blocks the IP addresses of the "naughty" hosts for a given amount of time. The "blocking" process relies on the packet filter functionalities by inserting to-be-blocked IP addresses into the firewall blocking list using PFCTL commands.

A simple web interface is provided for NIPS as shown below:

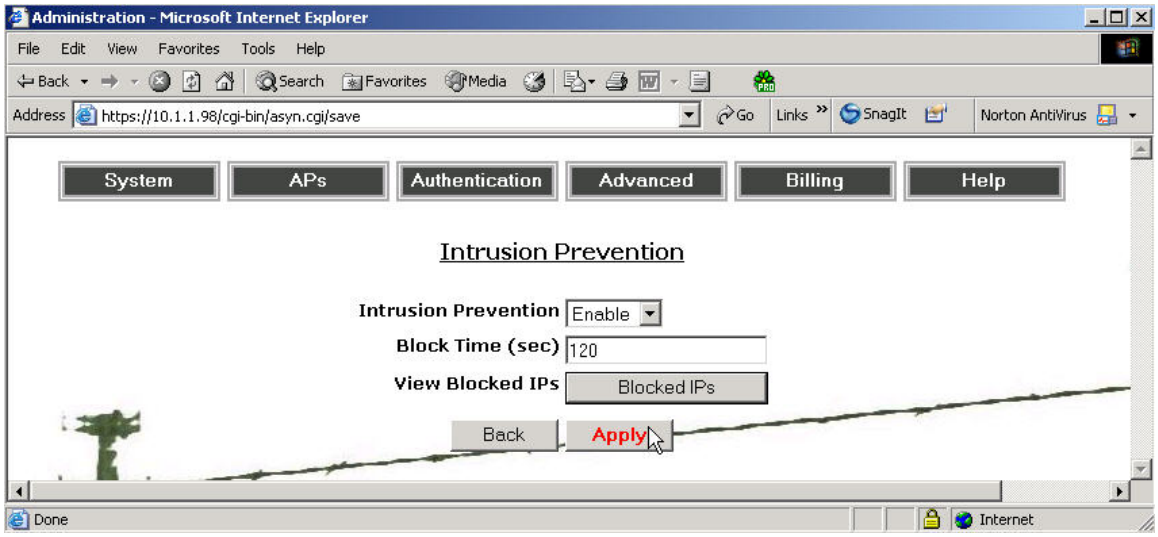


Figure 6: Intrusion Prevention Web Interface

You can choose specify how long the potential attacker should be blocked. Our NIPS will react to the attack immediately and block the IP address of the naughty hosts in seconds. A list of currently blocked IP address is provided and can be viewed from the web.

3. Wireless Attacks

a. Background

Wireless LAN security is receiving a large amount of attention these days. In addition to providing great mobility, it is also inexpensive and easy-to-use. However, the nature of wireless networks makes them attractive for intruders. Many intruders are only searching for free Internet access, and use a number of different probe tools to find it. Others are intent on gaining access to an enterprise network for malicious purposes – steal data, disrupt communications, damage files, etc. Moreover, after experiencing the benefits of wireless access, employees and executives of all types of companies naturally want to bring these benefits into the workplace – many times by plugging a consumer-grade access point into an Ethernet jack, and in effect extending that Ethernet port into the parking lot. This is the most serious form of security breach, because it has the ability to bypass all other security provisions.

Our intrusion detection and prevention system allows one to specify custom rules for detecting specific 802.11 frames, rogue access points, AdHoc networks, and Netstumbler like behavior. In order to accomplish this, our NIDS wireless rule engine has been augmented with support for a new "wifi" protocol. The remainders of the features are implemented as preprocessors that can be configured and tuned as desired according to the site of deployment.

b. "WiFi" Protocol Rules

Rules for detecting particular 802.11 frames are specified using the following syntax:

```
<action> wifi <src mac> -> <dst mac> (<rule options>)
```

RogueAP Preprocessor: The RogueAP preprocessor detects both rogue APs and AdHoc networks. The APs BSSIDs and channels they operate on are specified in the configuration file using the ACCESS_POINTS and CHANNELS variables. The following is an example:

```
# Single AP
var ACCESS_POINTS XX:XX:XX:XX:XX:XX
# Multiple APs
var ACCESS_POINTS [XX:XX:XX:XX:XX:XX, YY:YY:YY:YY:YY:YY, ....]
# Single channel
var CHANNELS X
# Multiple channels
var CHANNELS [X, Y, ...]
```

The preprocessor is activated by specifying the following line in the configuration file:

```
preprocessor rogue_ap: $ACCESS_POINTS, $CHANNELS, scan_flag [0 |

1], scan_timeout [num], expire_timeout [num]
```

The scan_flag toggles scaqnning of multiple channels. The scan_timeout specifies the time in seconds between channel scans. Those two are still under development. The expire_timeout specifies the time in seconds before a BSSID is removed from the list.

***AntiStumbler Preprocessor**:* The AntiStumbler preprocessor attempts to detect Netstumbler like traffic. It does this by keeping track of probe request frames sent with NULL SSID fields. The preprocessor is activated by specifying the following in the configuration file:

```
preprocessor antistumbler: probe_reqs [num], probe_period [num],

expire_timeout [num]
```

The probe_reqs specifies the number of probe requests that triggers an alert. The probe_period indicates the time period in seconds that NULL SSID probe request count is maintained. The expire_timeout specifies the time in seconds before a STA is removed from the stumbler list

Our Intrusion Detection system currently supports about fifty WiFi rules. As time goes on, more will be added.

Raw Data Management

1. User/Customer Data

Our system stores customer data and transaction data in a relational database that is built into the system. The standard SQL is used for the communication to the relational database. This opens up a broad range of applications that can possible be developed in the future. For example, open APIs in various languages such as C, Perl, Java could be implemented to allow direct access to the raw data. With such open APIs, customer data and transaction data can be exported into various formats such as XML, MS Excel format, CVS,…etc or directly transferred to other software systems such as RADIUS.

2. Backup data

Currently administrator can backup system configuration data, firewall rules, user data and role data. However, these backup data are not subject to version control. In the future, version control and comments could be allowed for backup data so that administrators can easily manage the backups and restore the system back to whatever previous states that had been saved.

3. Log files

Our log files are saved in the tcpdump format, which can later on be fed into various software applications (that accept the tcpdump format) to generate dynamic statistics for analyzing and reporting server traffic.

GUI Customization

Currently AWG-1000 allows some extent of customizations. For example, you can change the web page background picture, the company logo, the title of the Administrator and Client login pages. In the future, more customization such as navigation button's shape and style and UI text's font, size, color can be implemented as well to suite the need from different end-users computers (Windows, Mac, Linux, PDA, Xbox,..etc).

Radius Accounting

By default GNU Radius supports three types of accounting. Any additional accounting methods can be defined using extension mechanisms.

The accounting methods are applied to a request in a following sequence:

1. UNIX accounting : UNIX style utmp/wtmp accounting
2. Detailed request accounting
3. SQL accounting: Accounting to SQL server
4. Custom accounting

In this sequence, only UNIX accounting is obligatory; all other methods are applied only when enabled. If any accounting type in this sequence fails, the accounting is deemed to fail and all subsequent methods are not invoked.

1. UNIX Accounting

This accounting method is always enabled.

Radius keeps files `radutmp' and `radwtmp' in its logging directory and stores the accounting data there. The utilities radwho and radlast can be used to list information about users' sessions.

2. Detailed Request Accounting

Radius stores the detailed information about accounting packets it receives in files `radacct/nasname/detail', where nasname is replaced with the short name of the NAS from the `raddb/naslist' file. By default, this accounting type is always enabled, provided that `radacct' directory exists and is writable. To turn the detailed accounting off, use the detail statement in the `config' file.

The accounting detail files consist of a record for each accounting request. A record includes the timestamp and detailed dump of attributes from the packet, e.g.:

```
Fri Dec 15 18:00:24 2000
        Acct-Session-Id = "2193976896017"
        User-Name = "e2"
        Acct-Status-Type = Start
        Acct-Authentic = RADIUS
        Service-Type = Framed-User
        Framed-Protocol = PPP
        Framed-IP-Address = 11.10.10.125
        Calling-Station-Id = "+15678023561"
        NAS-IP-Address = 11.10.10.11
        NAS-Port-Id = 8
        Acct-Delay-Time = 0
        Timestamp = 976896024
        Request-Authenticator = Unverified

Fri Dec 15 18:32:09 2000
        Acct-Session-Id = "2193976896017"
        User-Name = "e2"
        Acct-Status-Type = Stop
        Acct-Authentic = RADIUS
        Acct-Output-Octets = 5382
        Acct-Input-Octets = 7761
        Service-Type = Framed-User
        Framed-Protocol = PPP
        Framed-IP-Address = 11.10.10.125
        Acct-Session-Time = 1905
        NAS-IP-Address = 11.10.10.11
        NAS-Port-Id = 8
        Acct-Delay-Time = 0
        Timestamp = 976897929
```

```
Request-Authenticator = Unverified
```

Notice that radiusd always adds two pseudo-attributes to detailed listings. Attribute Timestamp shows the UNIX timestamp when radiusd has received the request. Attribute Request-Authenticator shows the result of checking the request authenticator. Its possible values are:

Verified

The authenticator check was successful.

Unverified

The authenticator check failed. This could mean that either the request was forged or that the remote NAS and radiusd do not agree on the value of the shared secret.

None

The authenticator check is not applicable for this request type.

Notice also that the so-called *internal attributes* by default are not logged in the detail file. Internal attributes are those whose decimal value is greater than 255. Such attributes are used internally by radius and cannot be transferred via RADIUS protocol. Examples of such attributes are Fall-Through, Hint and Huntgroup-Name. See section 14.3 Radius Internal Attributes, for detailed listing of all internal attributes. The special attribute flag l (lower-case ell) may be used to force logging of such attributes (see section 5.2.4 ATTRIBUTE statement).

3. SQL Accounting

The SQL accounting method is enabled when Radius is configured with `--enable-sql' option and the `sqlserver' file in its configuration directory is properly set up. The GNU Radius 1.2 supports MySQL and PostgreSQL servers. It also supports ODBC, which can be used to build interfaces to another database management systems. With this accounting method enabled, radiusd will store the information about accounting requests in the configured SQL database. The accounting method is fully configurable: the Radius administrator defines both the types of requests to be accounted and the information to be stored into the database.

a. *“sqlserver” File Configuration*

The `raddb/sqlserver' file configures the connection to SQL server. The file uses simple line-oriented `keyword -- value' format. Comments are introduced by `#' character.

The `sqlserver' statements can logically be subdivided into following groups: *SQL Client Parameters*, configuring the connection between SQL client and the server, *Authentication Server Parameters*, *Authorization Parameters*, and *Accounting server parameters*.

b. *SQL Client Parameters*

These parameters configure various aspects of connection between SQL client and the server.

interface iface-type

Specifies the SQL interface to use. Currently supported values for iface-type are mysql and postgres. Depending on this, the default communication port number is set: it is 3306 for interface mysql and 5432 for interface postgres. Use of this statement is only meaningful when the package was configured with both `--with-mysql' and `--with-postgres' option.

server string

Specifies the hostname or IP address of the SQL server.

port number

Sets the SQL communication port number. It can be omitted if your server uses the default port.

login string

Sets the SQL user login name.

password password

Sets the SQL user password.

keepopen bool

Specify whether radiusd should try to keep the connection open. When set to no (the default), radiusd will open new connection before the transaction and close it right after finishing it. We recommend setting keepopen to yes for heavily loaded servers, since opening the new connection can take a substantial amount of time and slow down the operation considerably.

idle_timeout number

Set idle timeout in seconds for an open SQL connection. The connection is closed if it remains inactive longer that this amount of time.

C. Authentication Server Parameters

These parameters configure the SQL authentication. The general syntax is:

doauth bool

When set to yes, enables authentication via SQL. All auth_ keywords are ignored if doauth is set to no.

auth_db string

Specifies the name of the database containing authentication information.

auth_query string

Specifies the SQL query to be used to obtain user's password from the database. The query should return exactly one string value -- the password.

group_query string

Specifies the query that retrieves the list of user groups the user belongs to. This query is used when Group or Group-Name attribute appears in the LHS of a user's or hint's profile.

d. Authorization Parameters

These parameters define queries used to retrieve the authorization information from the SQL database. All the queries refer to the authentication database.

check_attr_query string

This query must return a list of triplets:

attr-name, attr-value, opcode

The query is executed before comparing the request with the profile entry. The values returned by the query are added to LHS of the entry. opcode here means one of valid operation codes: `=', `!=', `<', `>', `<=', `>='.

reply_attr_query string

This query must return pairs:

attr-name, attr-value

The query is executed after a successful match, the values it returns are added to the RHS list of the matched entry, and are therefore returned to the NAS in the reply packet.

e. Accounting Parameters

To perform the SQL accounting radiusd needs to know the database where it is to store the accounting information. This information is supplied by the following statements:

doacct bool

When set to yes enables SQL accounting. All acct_ keywords are ignored if doacct is set to no.

acct_db string

Specifies the name of the database where the accounting information is to be stored.

Further, radiusd needs to know which information it is to store into the database and when. Each of five accounting request types has a SQL query associated with it. Thus, when radius receives an accounting request, it determines the query to use by the value of Acct-Status-Type attribute.

Following statements define the accounting queries:

acct_start_query string

Specifies the SQL query to be used when *Session Start Packet* is received. Typically, this would be some INSERT statement.

acct_stop_query string

Specifies the SQL query to be used when *Session Stop Packet* is received. Typically, this would be some UPDATE statement.

acct_stop_query string

Specifies the SQL query to be executed upon arrival of a *Keepalive Packet*. Typically, this would be some UPDATE statement.

acct_nasup_query string

Specifies the SQL query to be used upon arrival of an *Accounting Off Packet*.

acct_nasdown_query string

Specifies the SQL query to be used when a NAS sends *Accounting On Packet*.

None of these queries should return any values.

Full Wireless Switch Functions

In addition to the wireless gateway features described above, AWG-1000 also serves as a full wireless switch that performs a number of functions that enhances wireless security, RF management, mobility, QoS, etc.

1. RF Management

The staple of a wireless switch is its ability to manage the radio frequency; AWG-1000 is equipped with a strong set of RF management functions:

- Automatic and dynamic channel selection
- Dynamic power control
- Coverage void detection and correction
- Interference detection and correction
- RF Planning software allows modeling, planning and placement of APs
- Centralized and distributed calibration automates AP deployment and provides optimal coverage
- Self-healing capabilities for APs and AP-to-switch connectivity minimizes network downtime
- Load balancing and automated user-AP assignment

2. Mobility

The highly efficient and reliable roaming capability of AWG-1000 allows mobility for all users and applications. Secure IP roaming allows enterprisewide mobility across different subnets, APs, switches. Also, proxy DHCP enables VPN tunnel persistence as user roam across subnets.

3. QoS

As a wireless switch, the AWG-1000 is well prepared for quality-sensitive applications, such as voice. There are a number of features that provide the users a well managed environment:

- Fast handoffs between APs for VoIP mobility
- Stateful flow classification for prioritization of VoIP and streaming media
- Bandwidth management to enforce usage limits
- 802.1p/Diffserv support for prioritizing traffic across the wired and wireless networks

4. System Management

Programmable platform allows easy deployment of upgrades and new wireless applications. Automated updating capability relieves the system administration staff the mundane and error-prone work of constantly pushing changes out to a large and complex set of wireless units.

Multiple AWG-1000 switches distributed in branch or regional offices can be centrally managed from a master AWG-1000 Wi-Fi switch. All switch configuration and user policies can be defined from the master switch and automatically propagated to AWG-1000 Wi-Fi switches in remote locations. From a master AWG-1000 Wi-Fi switch, administrators can manage, secure and control branch office RF environments, capture wireless traffic and remotely troubleshoot problems.

Also, there is the ability to centrally upgrade and manage access points.

5. Security

In addition to the rich set of security functions that AWG-1000 has, there are more features that particularly address the wireless switch operations. These include:

- Rouge AP detection and containment
- Wireless Intrusion detection
- Exclusion lists based on a number of parameters

For More Information

- For a complete features list of AWG-1000, please visit http://www.wiborne.com