Implementation of Wireless VoIP System based on VPN

Jung-Yong Park, Dae-Hyun Ryu, *Department of IT , Hansei University*

Abstract—VoIP is vulnerable to attack since it uses the Internet to which many people connect simultaneously. In this paper, we designed and implemented a Wireless VoIP + VPN system with which secure telephone calls are possible using the open project SIP VoIP Gateway, 'Asterisk' and 'OpenVPN'.

Index Terms—VoIP, VPN, SIP, Wireless, Security, WiFi

I. INTRODUCTION

VoIP service is based on Internet technologies. With traditional PSTN, it is not trivial to eavesdrop because the connection is established in the form of a 1:1 circuit. However, VoIP isn't as secure since it uses the Internet to which many people connect simultaneously. In the case of a wireless Internet environment, security may be further compromised due to A.P. vulnerability.

In this paper, we designed and implemented a Wireless VoIP + VPN system with which secure telephone calls are possible using the open project SIP VoIP Gateway 'Asterisk' and 'OpenVPN'. With Wireless VoIP + VPN , we can save money and improve the level of security by integrating voice data with streaming and encryption. In other words, we can get total security by integrating Wireless VoIP and VPN.

II. VoIP Security

A. VoIP

We used 'Asterisk'[3] which is an open project for VoIP communication in wireless LAN environments. For client programming, we used QT/Embedded which is an embedded LINUX GUI library. We use SIP[1] for configuration of mutual connections and RTP[2] when transmitting voice packets.

The Session Initiation Protocol (SIP) is an application-layer control (signaling) protocol for creating, modifying, and terminating sessions with one or more participants. It can be used to create two-party, multiparty, or multicast sessions that include Internet telephone calls, multimedia distribution, and multimedia conferences. (cit. RFC 3261).

Manuscript received February 14, 2008.

Dae-Hyun Ryu is with the Department of IT, Hansei University, Cunpo-Si, Gyunggi, 435-742, Korea (Tel: +82-11-450-0132, Fax: +82-31-450-5172, Email: dhryu@hansei.ac.kr)

SIP has the following characteristics:

- Transport-independent, because SIP can be used with UDP, TCP, ATM & so on.
- Text-based, allowing for humans to read SIP messages.

The Real-time Transport Protocol (or RTP) defines a standardized packet format for delivering audio and video over the Internet. It was originally designed as a multicast protocol, but has since been applied in many unicast applications. It is frequently used in streaming media systems (in conjunction with RTSP) as well as videoconferencing and push to talk systems (in conjunction with H.323 or SIP), making it the technical foundation of the Voice over IP industry.

VoWLAN (Voice over WLAN) is a method of sending voice information in digital form over a wireless broadband network. Essentially, VoWLAN is VoIP delivered through wireless technology. The technology is sometimes called "VoWi-Fi" or "Wi-Fi VoIP" because it uses the IEEE 802.11 set of specifications (informally known collectively as Wi-Fi) for transporting data over wireless local area networks and the Internet.

B. Security Vulnerability of Wireless VoIP

Most wireless local area networks (WLANs) today are built on Wi-Fi technologies, i.e., those based on the IEEE 802.11 wireless standard. IEEE 802.11 is a set of standards for wireless local area network (WLAN) computer communication, developed by the IEEE LAN/MAN Standards Committee (IEEE 802) in the 5 GHz and 2.4 GHz public spectrum bands.

Due to security reasons, 802.11 employs the wired equivalent privacy (WEP) protocol. WEP was intended to give the wireless data-link a level of security similar to that of naturally-built in wires and optical links. WEP's goals are to provide access control, data confidentiality and data integrity. It does this by using symmetric key mechanisms. With WEP, all devices must have a secret WEP key entered into them by the network administrator. This is usually done manually. By now, it is well known that WEP is extremely vulnerable and can not be counted on to defend against even casual attackers since there are scripts available online that can defeat WEP in a matter of minutes. While there are many proposed fixes for WEP, most of them are not applicable until the next generation of wireless hardware due to their increased computational requirements.

A service set identifier, or SSID, is a name used to identify the particular 802.11 wireless LANs to which a user wants to attach. A client device will receive

broadcast messages from all access points within range advertising their SSIDs, and can choose one to connect to based on pre-configuration, or by displaying a list of SSIDs in range and asking the user to select one. So, anyone can gain access to a wireless AP with some tools such as NetStumbler which can detect Wireless LANs using the 802.11b, 802.11a and 802.11g WLAN standards since the SSID is broadcasted to the users of the AP.

VoWLAN can also easily eavesdropped by packet sniffing tools like Ethereal. Eavesdroppers have the ability to capture plain and cipher text and get shared keys using a protocol analyzer. So, it is susceptible to man-in-the-middle attacks. An attack targeting ARP (Address Resolution Protocol) is a typical man-in-middle attack. This attack exploits a critical point of ARP that enables hackers to impersonate the computer that the user wants to communicate with.

III. Secure Protocols

A. SSL (Secure Socket Layer)

Secure Sockets Layer (SSL), are cryptographic protocols that provide secure communications on the Internet for such things as web browsing, e-mail, Internet faxing, instant messaging and other data transfers. There are slight differences between SSL and TLS, but the protocol remains substantially the same.

The SSL protocol allows applications to communicate across a network in a way designed to prevent eavesdropping, tampering, and message forgery. SSL provides endpoint authentication and communications privacy over the Internet using cryptography. Typically, only the server is authenticated (i.e., its identity is ensured) while the client remains unauthenticated; this means that the end user (whether an individual or an application, such as a Web browser) can be sure with whom they are communicating. Fig. 1 shows SSL Protocol Stack

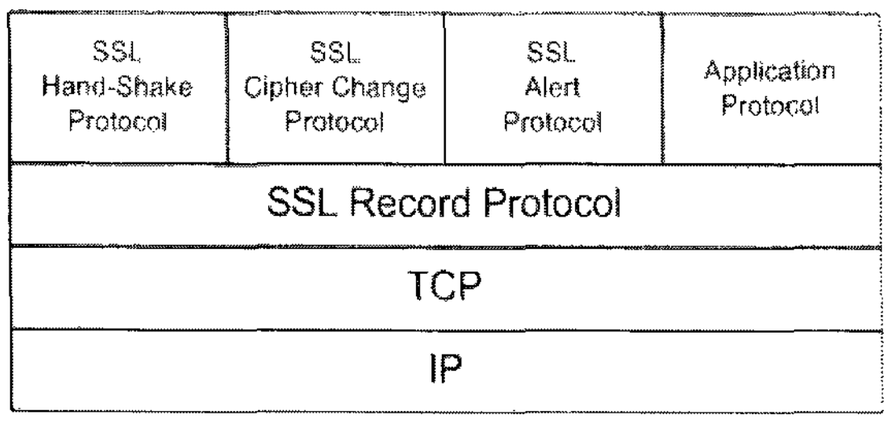


Fig. 1 SSL Protocol Stack

A SSL client and server negotiate a stateful connection by using a handshaking procedure. During this handshake, the client and server agree on various parameters used to establish the connection's security. The handshake begins when a client connects to a SSL-enabled server requesting a secure connection, and presents a list of supported ciphers and hash functions. From this list, the server picks the strongest cipher and hash function that it also supports and notifies the client of the decision. The server sends back its identification in the form of a digital certificate. The certificate usually contains the server name, the trusted certificate authority (CA), and the server's public encryption key. The client may contact the server that issued the certificate (the trusted CA as above) and confirm that the certificate is authentic before proceeding. In order to generate the session keys used for the secure connection, the client encrypts a random number with the server's public key, and sends the result to the server. Only the server can decrypt it (with its private key): this is the one fact that makes the keys hidden from third parties, since only the server and the client have access to this data. From the random number, both parties generate key material for encryption and decryption. This concludes the handshake and begins the secured connection, which is encrypted and decrypted with the key material until the connection closes.

B. IPsec

IPsec protocols operate at the network layer, layer 3 of the OSI model. Other Internet security protocols in widespread use, such as SSL, TLS and SSH, operate from the transport layer up (OSI layers 4 - 7). This makes IPsec more flexible, as it can be used for protecting layer 4 protocols, including both TCP and UDP, the most commonly used transport layer protocols. IPsec has an advantage over SSL and other methods that operate at higher layers: An application needn't be designed to use IPsec, whereas the ability to use SSL or another higher-layer protocol must be incorporated into the design of an application [6]. IPsec is implemented by a set of cryptographic protocols for (1) securing packet flows, (2) mutual authentication and (3) establishing cryptographic parameters.

The IP security architecture uses the concept of a security association as the basis for building security functions into IP. A security association is simply the bundle of algorithms and parameters (such as keys) that is being used to encrypt and authenticate a particular flow in one direction. Therefore, in normal bi-directional traffic, the flows are secured by a pair of security associations. The actual choice of encryption and authentication algorithms (from a defined list) is left to the IPsec administrator.

In order to decide what protection is to be provided for an outgoing packet, IPsec uses the security parameter index (SPI), an index to the security association database (SADB), along with the destination address in a packet header, which together uniquely identify a security association for that packet. A similar procedure is performed for an incoming packet, where IPsec gathers decryption and verification keys from the security association database.

Two protocols have been developed to provide

packet-level security for both IPv4 and IPv6: The IP Authentication Header provides integrity and authentication and non-repudiation, if the appropriate choice of cryptographic algorithms is made. The IP Encapsulating Security Payload provides confidentiality, along with optional (but strongly recommended) authentication and integrity protection.

The AH is intended to guarantee connectionless integrity and data origin authentication of IP datagrams. Further, it can optionally protect against replay attacks by using the sliding window technique and discarding old packets.

The ESP protocol provides origin authenticity, integrity, and confidentiality protection of a packet. ESP also supports encryption-only and authentication-only configurations, but using encryption without authentication is strongly discouraged because it is insecure.[1][2][3]. Unlike AH, the IP packet header is not protected by ESP. (Although in tunnel mode ESP, protection is afforded to the whole inner IP packet, including the inner header; the outer header remains unprotected.) ESP operates directly on top of IP, using IP protocol number 50.

C. OpenVPN

OpenVPN allows peers to authenticate to each other using a pre-shared secret key, certificates, or username/password. It uses the OpenSSL encryption library extensively, as well as the SSLv3/TLSv1 protocol. It is available on Solaris, Linux, OpenBSD, FreeBSD, NetBSD, Mac OS X, and Windows 2000/XP. It contains many security and control features. It is not a "web-based" VPN, and is not compatible with IPsec or any other VPN package. The entire package consists of one binary executable for both client and server connections, an optional configuration file, and one or more key files depending on the authentication method used.

OpenVPN uses the OpenSSL library to provide encryption of both the data and control channels. It lets OpenSSL do all the encryption and authentication work, allowing OpenVPN to use all the ciphers available in the OpenSSL package. It can also use the HMAC packet authentication feature to add an additional layer of security to the connection (referred to as an "HMAC Firewall" by the creator). It can also use hardware acceleration to get better encryption performance.

OpenVPN is a full-featured SSL VPN solution which can accommodate a wide range of configurations, including remote access, site-to-site VPNs, WiFi security, and enterprise-scale remote access solutions with load balancing, failover, and fine-grained access-controls.

OpenVPN implements OSI layer 2 or 3 secure network extension using the industry standard SSL/TLS protocol, supports flexible client authentication methods based on certificates, smart cards, and/or 2-factor authentication, and allows user or group-specific access control policies using firewall rules applied to the VPN virtual interface. OpenVPN is not a web application proxy and does not operate through a web browser.

IV. VPN + Wireless VoIP

In this paper, we designed and implemented a secure wireless VoIP system utilizing 'OpenVPN'

A. Hardware

We use a PC with Intel's Pentium Ⅳ 3,0GHz CPU with 1024MB RAM as described in Fig. 2. It can be connected and interoperated with PSTN using a Digium TMD400P PCI card. The TDM400P card consists of two FXO modules. One is connected to a typical telephone and VoIP network, the other is for PSTN.

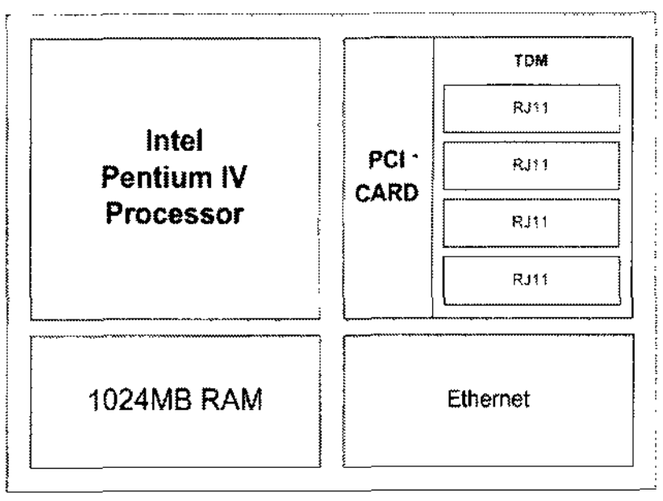


Fig. 2 Block Diagram of Server H/W.

We used a Bookdoo IN-DVK-P255B embedded board as a client (Fig. 3) with Intel's XScale PXA255 CPU, 128MB SDRAM, 64MB Flash Memory and PCMCIA interface. We used Orinoco PCMCIA Wireless LAN Card for IEEE 801.11b connectivity.

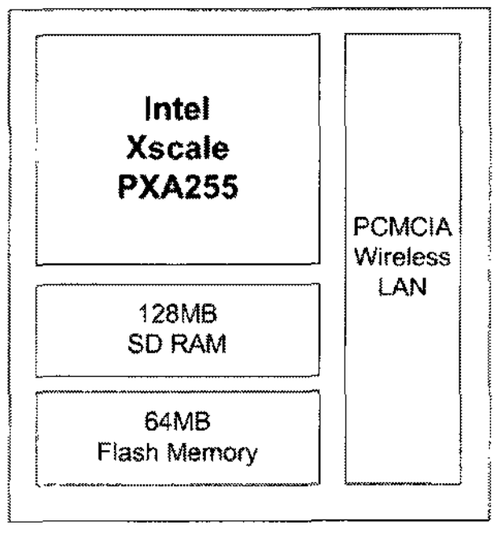


Fig. 3 Block Diagram of Client H/W.

B. Software

In this paper, we designed and implemented VPN based VoIP using the open project SIP VoIP Gateway 'Asterisk' and 'OpenVPN'. We use Linux Kernel 2.6.18 and ANSI-C with GCC complier.

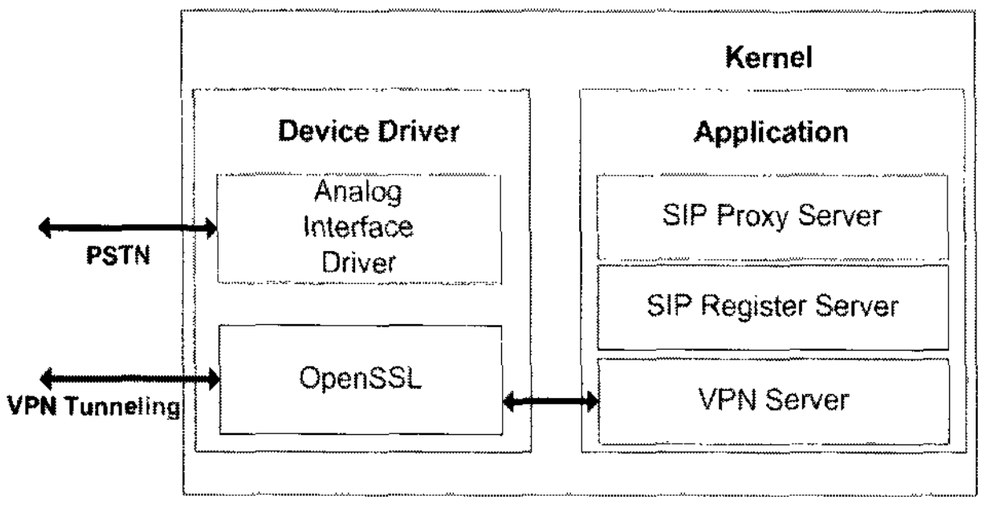


Fig. 4 Block Diagram of Server S/W.

As per Fig. 4, clients connect to the server and establish VPN connections between the clients, then transmit encrypted packets using a tunnel interface.

C. Network Configuration

In this paper, we configured the networks as follows

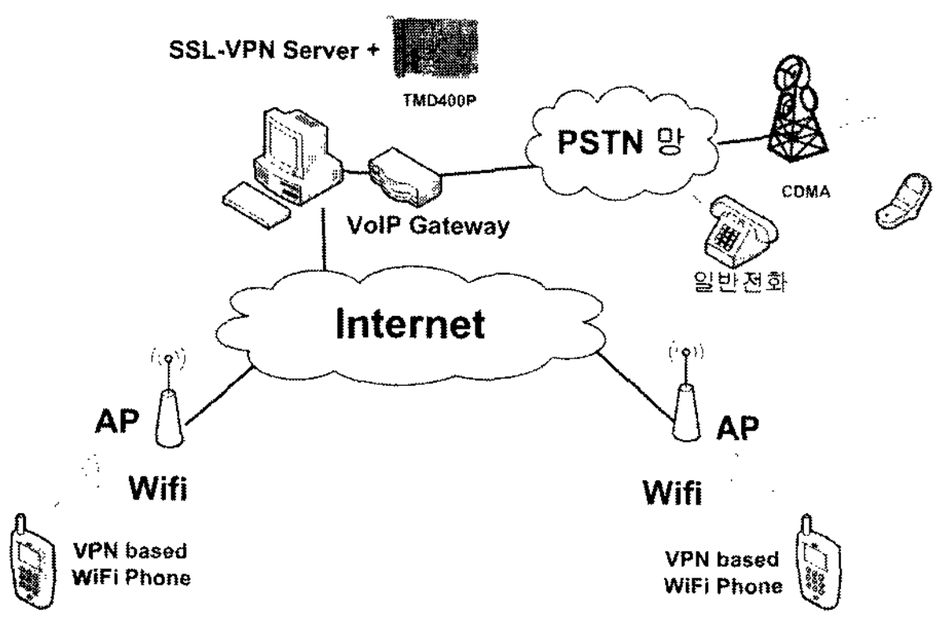


Fig. 5 Network Configuration.

Ⅴ. Performance Evaluation

We configured the networks as per Fig. 5 and used the embedded boards with WiFi as a WiFi phone. The server operates as a SIP Proxy Server and the codec for voice compression is G.711. The server also works as a VPN-Server and is configured to connect authenticated users with the Tunnel-Interface. We can use PSTN networks using the server FXO terminals.

We compared the voice quality between a typical VoIP call and our VoVPN call. We measured the Round Trip Time which is the time between sending and receiving voice packets to get the end-to-end packet delay time. We checked that the difference between the end-to-end packet delay time of a typical VoIP call and our VoVPN call is less than a maximum of 20ms. We hypothesize that the greater part of the delay time is used to encapsulate packets for VPN tunneling. It is less than the ITU-T suggested

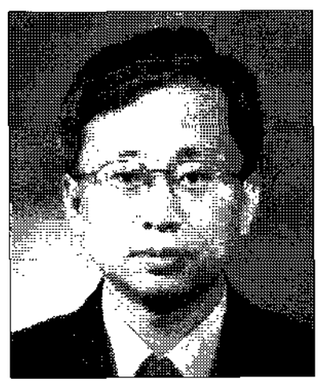
Ⅵ. CONCLUSIONS

In this paper, we designed and implemented a Wireless VoIP + VPN system with which secure telephone calls are possible using the open project SIP VoIP Gateway 'Asterisk' and 'OpenVPN'. With Wireless VoIP + VPN, we can save money and improve the level of security by integrating voice data streaming and encryption. We measured the end-to-end packet delay time of a typical VoIP call and our VoVPN call for performance evaluation. The difference between our system and a typical VoIP call is under 20ms and less than the ITU-T suggested end-to-end packet delay time limit of 150ms ~ 200ms

REFERENCES

[1] RFC 3261, M. Handley, H. Schulzrinne, E. Schooler, J. Resenberg "SIP: Session Initiation Protocol". *IETF*, Jun 2002.
[2] RFC 1890, "RTP Profile for Audio and Video Conferences with Minimal Control", IETF, Jan. 1996.
[3] http://asterisk.org/about. "What is Asterisk".
[4] "Review of Wireless Internet Security Technologies" , KISA 2001. 11
[5] Stephen A. Thomas, *SSL and TLS Essentials*, Wiey, 2000
[6] J.H.Nahm, "Compare & Implementation of IPsec VPN vs SSL VPN", Semyung Uni., 2004
[7] http://openvpn.net/. "OpenVPN"
[8] http://www.ethereal.com/introduction.html, "Features".
[9] TIA/EIA/TSB116, Voice Quality Recommendations for IP Telephony, Mar 2001.

Jung-Yong Park
Jung-Yong Park received his B.S. degree in IT from Hansei University in 2008. Shortly following graduation, he started at BeyonWiz, where he worked as a junior member of the technical staff. His research interests are in the area of wireless networks, VoIP, security and multimedia systems.



Dae-Hyun Ryu
Prof. Dae-Hyun Ryu received his B.S. degree, M.S. and Ph.D. degrees in Electrical and Electronic Engineering from Busan National University in 1983, 1985 and 1997, respectively. From 1987 to 1998, he worked at ETRI, where he worked as senior member of technical staff. In 1998, he joined the department of IT, Hansei University, Korea. His research interests are in the area of digital image processing, digital watermarking and information security system design.