CS 393 – Network Security

Nasir Memon – Polytechnic University
Module 13 – Virtual Private Networks
Course Logistics

- HW due Monday.
- HW 6 posted. Due in a week.
- Questions regarding homework are best answered by Vikram – vikram@vip.poly.edu
- Apply for DoD scholarships. Deadline postponed to May 6th!
Where do we provide security?

- Firewall
- Application layer
  - S-MIME, PGP, Kerberos etc.
- Transport layer
  - SSL, TSL.
- Network layer
  - VPN, IPSEC
Public and Private Networks

- What is a Public Network?
  - Examples - Internet, PSTN
  - Insecure, unreliable
  - Cheap. Ubiquitous.

- What is a Private Network?
  - Examples - LANs, Intranet
  - Secure, reliable
  - Expensive. Limited availability and access

- Virtual Private Networks try to get the best of both worlds!
Dedicated Connectivity Solutions
Connectivity Using Internet

- Headquarters
  New York, USA
- Sales Offices
  Latin America
  South East Asia
  Europe
  USA
- Manufacturing
  Taipei, Taiwan
- Suppliers
  Singapore
  Sao Paulo
  Dallas
- Customers
  North America
  Europe
  South East Asia
  Latin America
- Mobile Worker
Internet is the ubiquitous network.
What is a VPN?

- A Virtual Private Network (VPN) is a private network constructed within a public network infrastructure, such as the global Internet.
- A VPN connects the components and resources of one network over another network.
- Usually, this is done by a combination of tunneling, encryption, authentication, and access control technologies and services used to carry traffic over the Internet, a managed IP network or a provider's backbone.
Virtual Private Network

Transit Internetwork

Virtual Private Network

Logical Equivalent
Why VPN? Driving Forces

- Dispersed and Mobile Workforce
- High Cost of Private Networks
- Virtual Private Networking
- On Line Interaction with Customers and Suppliers
- Consolidation and Simplification of the Interface
VPNs provide remote access to corporate resources over the public Internet, while maintaining privacy of information.

Rather than making a leased line, long distance (or 1-800) call to a corporate or outsourced Network Access Server (NAS), the user first calls a local ISP NAS phone number.

Using the local connection to the ISP, the VPN software creates a virtual private network between the dial-up user and the corporate VPN server across the Internet.
Using a VPN to connect a remote client to a private LAN

![Diagram showing a VPN connection between a remote client, an ISP, and a corporate hub.]
Using VPN to Connect a Branch Office to a Corporate LAN

- Rather than using an expensive long-haul dedicated circuit between the branch office and the corporate hub, both the branch office and the corporate hub routers can use a local dedicated circuit and local ISP to connect to the Internet.

- The VPN software uses the local ISP connections and the public Internet to create a virtual private network between the branch office router and corporate hub router.
Using a VPN to connect two remote sites
VPN for Connecting Computers Over an Intranet

- In some corporate internets, the departmental data is so sensitive that the department's LAN is physically disconnected from the rest of the corporate internet.

- While this protects the department's confidential information, it creates information accessibility problems for those users not physically connected to the separate LAN.

- VPN can be used to provide access to secured LAN from corporate intranet.
Using a VPN to connect to two computers on the same LAN
VPN Requirements

- **User Authentication.** VPN must verify a user's identity and restrict VPN access. Audit and accounting records to show who accessed what information and when.

- **Address Management.** VPN must assign client's address on private net and ensure that addresses are kept private.

- **Data Encryption.** Data carried on the public network must be rendered unreadable to unauthorized clients on the network.

- **Key Management.** VPN must generate and refresh encryption keys for the client and server.

- **Multiprotocol Support.** VPN must be able to handle common protocols used in the public network. These include Internet Protocol (IP), Internet Packet Exchange (IPX), and so on.
Tunneling

- VPNs allow the user to tunnel through the Internet or another public network in a manner that lets the tunnel participants enjoy the same security and features available in private networks.
- The secure connection across the internetwork appears to the user as a private network communication—despite the fact that this communication occurs over a public internetwork—hence the name Virtual Private Network.
Tunneling
Tunneling

- Tunneling is a method of using an internetwork infrastructure to transfer data from one network over another network.
- The data to be transferred (or payload) can be the frames (or packets) of another protocol.
- Instead of sending a frame as it is produced by the originating node, the tunneling protocol encapsulates the frame in an additional header.
- The additional header provides routing information so that the encapsulated payload can traverse the intermediate internetwork.
Tunneling

- The encapsulated packets are then routed between tunnel endpoints over the internetwork.
- The logical path through which the encapsulated packets travel through the internetwork is called a tunnel.
- Once the encapsulated frames reach their destination on the internetwork, the frame is unencapsulated and forwarded to its final destination.
- Note that tunneling includes this entire process (encapsulation, transmission, and unencapsulation of packets).
Tunneling Protocols

- Generic Routing Encapsulation (GRE), as defined in RFCs 1701/1702, is used by a wide variety of tunneling protocols.
- The Point-to-Point Tunneling Protocol (PPTP), created by Microsoft and Ascend Communications, is an extension of Point-to-Point protocol (PPP) for Windows and Netware client/server environments.
- The Ascend Tunnel Management Protocol (ATMP), defined as RFC 2107, implements both GRE and PPTP for tunneling IP, IPX, NetBIOS and NetBEUI traffic.
Tunneling Protocols

- Layer-2 Forwarding (L2F) is a tunneling protocol created by Cisco Systems.
- The Layer-2 Tunneling Protocol (L2TP) is proposed standard to combine best features of L2F and PPTP.
- Data Link Switching (DLSw), defined by IBM and now an industry standard, encapsulates SNA traffic in IP.
- Mobile IP is designed to tunnel IP within IP for individuals traveling away from their "home" network.
- IPsec supports tunneling with or without encryption.
Why So Many VPN Solutions?

- For some companies, a VPN is a substitute for remote-access servers.
- For others, a VPN may consist of traffic over the Internet between protected LANs.
- The protocols that have been developed for VPNs reflect this dichotomy.
- PPTP, L2F and L2TP are largely aimed at dial-up VPNs
- IPSec's main focus has been LAN-to-LAN solutions.
PPTP

- From Microsoft
- Widely supported standard
- Follows Client Server model
- Based on PPP
What is PPTP?

- Simply: PPP in IP
  - Traditional PPP dial up frames are encapsulated in IP
  - PPP is ubiquitous dial up standard
  - PPP is multi-protocol, extensible, and authenticated
Review of PPP

- PPP provides three things:
  - A framing method that unambiguously delineates the end of one frame and the start of the next one. The frame format also handles error detection.
  - A link control protocol for bringing up lines, testing them, negotiating options, and bringing them down gracefully when no longer needed. This protocol is called Link Control Protocol (LCP).
  - A way to negotiate network layer options in a way that is independent of the network layer protocol to be used. The method chosen is to have a different Network Control Protocol (NCP) for each network layer supported.
Review of PPP

- Typical connection setup:
  - Home PC Modem calls Internet
  - Provider's router: sets up physical link
  - PC sends series of LCP packets
  - Select PPP (data link) parameters
  - Authenticate
  - PC sends series of NCP packets
  - Select network parameters
    - E.g., Get dynamic IP address
  - Transfer IP packets
## Review of PPP

### Diagram

- **Carrier detected** → **Establish** → **Authenticate** → **Network**
- **Both sides agree on options** → **Authenticate** → **Open** → **NCP configuration**
- **Authentication successful** → **Network**
- **Failed** → **Terminate** → **Dead**
- **Carried dropped** → **Terminate** → **Dead**
- **Done** → **Open** → **NCP configuration**

### Table

<table>
<thead>
<tr>
<th>Bytes</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1 or 2</th>
<th>Variable</th>
<th>2 or 4</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flag</td>
<td>01111110</td>
<td>Address</td>
<td>11111111</td>
<td>Control</td>
<td>00000011</td>
<td>Protocol</td>
<td>Payload</td>
</tr>
<tr>
<td>Checksum</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Flag</td>
<td>01111110</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

4/9,11/02 Module 13 - VPN's
Review of PPP

- PPP handles error detection, supports multiple protocols, allows IP addresses to be negotiated at connection time, permits authentication. It is used on both dial-up lines and router to router leased lines.
Review of PPP

- PPP Suite:
- LCP
- NCPs - IPCP, IPXCP, ATCP, etc.
- CCP
- ECP
- CHAP
- PAP

<table>
<thead>
<tr>
<th>Name</th>
<th>Direction</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configure-request</td>
<td>I → R</td>
<td>List of proposed options and values</td>
</tr>
<tr>
<td>Configure-ack</td>
<td>I ← R</td>
<td>All options are accepted</td>
</tr>
<tr>
<td>Configure-nak</td>
<td>I ← R</td>
<td>Some options are not accepted</td>
</tr>
<tr>
<td>Configure-reject</td>
<td>I ← R</td>
<td>Some options are not negotiable</td>
</tr>
<tr>
<td>Terminate-request</td>
<td>I → R</td>
<td>Request to shut the line down</td>
</tr>
<tr>
<td>Terminate-ack</td>
<td>I ← R</td>
<td>OK, line shut down</td>
</tr>
<tr>
<td>Code-reject</td>
<td>I ← R</td>
<td>Unknown request received</td>
</tr>
<tr>
<td>Protocol-reject</td>
<td>I ← R</td>
<td>Unknown protocol requested</td>
</tr>
<tr>
<td>Echo-request</td>
<td>I → R</td>
<td>Please send this frame back</td>
</tr>
<tr>
<td>Echo-reply</td>
<td>I ← R</td>
<td>Here is the frame back</td>
</tr>
<tr>
<td>Discard-request</td>
<td>I → R</td>
<td>Just discard this frame (for testing)</td>
</tr>
</tbody>
</table>
The CHAP Process

Challenge = Session ID, Challenge String

Response = MD5 Hash(Session ID, Challenge String, User Password), User Name
PPTP Details

- **Tunnel**
  - Between PPTP FEP or Client, and PPTP server
  - Control and Data Channel
  - Many PPP sessions carried

- **Control Channel**
  - TCP based (IANA port 1723)
  - Session establishment, tear down, and management

- **Data Channel**
  - Enhanced GRE encapsulation (Ethertype 0x880B, IP Protocol ID 47)
  - Flow control for congestion and link feedback
Conceptual Model of PPTP

Remote Users

Any PPP client PC

PSTN
ISDN

PPTP-enabled client PC

Internet Svc Provider
Access Server
Front-Ends

Internet

PPTP

PPP(IP)

IP(PPP(IP))

POP B

Corporate Intranets

ABC Corp.

IP, IPX, or

XYZ Corp.

IP, IPX, or

Any PPP client PC

PPTP-enabled client PC

IP(PPP(IPX))
PPTP Components

- Generally, there are three computers involved in every PPTP deployment:
  - a PPTP client
  - a network access server
  - a PPTP server

- **Note:** You do not need the network access server in order to create a PPTP tunnel when using a PPTP client connected to a LAN to connect to a PPTP server connected to the same LAN.
The PPTP Tunnel
A Typical PPTP Data Packet

<table>
<thead>
<tr>
<th>Media</th>
<th>IP</th>
<th>GRE</th>
<th>PPP</th>
<th>PPP Payload</th>
</tr>
</thead>
</table>

4/9,11/02
PPTP Encapsulation

Diagram showing the encapsulation process from the remote client to the target network through a network access server and tunnel infrastructure.
GRE

Standard IP Header
(Using Home and Foreign Agents’ IP Address for Source/Destination)

GRE Header
- **Protocol Type** of Payload Packet
- **Checksum** (optional)
- **Key** to Identify/Authenticate Actual Source
- **Sequence Number** of Packet in Series Transmitted
- **Source Routing** (optional)

Payload
(Original Packet, Complete with Original Header)
PPTP Architecture Overview

- Secure communication created using PPTP typically involves three processes. Each requires successful completion of the previous process.

- **PPP Connection and Communication.** A PPTP client uses PPP to connect to an ISP by using a standard telephone line or ISDN line.

- **PPTP Control Connection.** Using the connection to the Internet established by the PPP protocol, the PPTP protocol creates a control connection from the PPTP client to a PPTP server on the Internet. This connection uses TCP and is called a PPTP tunnel.

- **PPTP Data Tunneling.** Finally, PPTP creates IP datagrams containing encrypted PPP packets which are sent through PPTP tunnel to the PPTP server. The PPTP server disassembles the IP datagrams and decrypts the PPP packets, and then routes the decrypted packets to the private network.
PPTP Authentication

- 3 Authentication options
  - cleartext passwords (like PAP)
  - hashed passwords
  - MS-CHAP (or “Microsoft Authentication”)

- Link-layer encryption available, if you pick “Microsoft Authentication”
LAN Manager Hash

- Developed by Microsoft for OS/2
- Used wherever network authentication is needed

Algorithm:
- Truncate/Pad password to 14 characters
- Upcase password
- Split the password into two 7-byte halves
- Using DES, build \( \{c\}k_1 + \{c\}k_2 \), \( c \) a constant
Flaws in the LAN Manager hash

- This algorithm is very vulnerable to dictionary attacks
  - Passwords less than 7 characters will result in a hash whose last 8 bytes are known
- And we can still brute-force each half of the hash separately if the password’s longer
- Password space is much smaller than DES keyspace - There’s no salt
  - So we can build up a dictionary of encrypted passwords
NT hash

- Later systems (NT and 95) calculate MD4(unicode password)
- ... but there’s still no salt
- and the password space is still small
- and most applications need to calculate both NT hash and LMHash for backwards-compatibility
MS-CHAP

- C->S challenge, please?
- S->C 8-byte challenge
- C calculates NT and LM hashes, pads to 21 bytes
- C divides NT and LM hash into 3 7-byte keys each and uses them to encrypt the challenge
- C->S \{challenge\}k1 + \{challenge\}k2 + \{challenge\}k3, preferred hash
- Both hashes are always sent
Problems with MS-CHAP

- Protocol is vulnerable to flaws in LM hashes
- Many ways to speed up key searches:
  - Null padding means the third key only has $2^{15}$ possibilities - this gets us 2 key bytes for cheap.
  - Once we’ve got these, we guess the middle keys, encrypt the LAN Manager constant under our guess and check just the last two bytes against them.
  - Finally, we brute-force the first 7 bytes
Microsoft Point-to-Point Encryption

- After authentication, MS PPTP does link-layer encryption
- 40 or 128-bit RC4
  - Key is generated by either hashing the LM Hash using SHA1 and truncating (40-bit mode), SHA1 on the concatenation of the 64-bit nonce and the 64-bit Windows NT hash (64-bit mode)
  - Key is regenerated every 256 packets by hashing the preceding key and the first key
- Key can be recovered by password cracking, even if it is 128 bit!
Getting the Encryption Key

- With 40 bit RC4, because there is no salt, an attacker can precompute a dictionary of ciphertext PPP headers, and then quickly look-up a given ciphertext in this dictionary.

- Moreover, the same 40-bit RC4 key is generated every time the same user initializes the PPTP protocol. Since RC4 is an output-feedback mode cipher, one can break the encryption from the ciphertext from two sessions.

- Same key is used for communications in both directions!! This automatically gives you above.
MS-CHAP v2

1. Client requests a login challenge from the Server.
2. The Server sends back a 16-byte random challenge.
3a. The Client generates a random 16-byte number, called the Peer Authenticator Challenge."
3b. The Client generates an 8-byte challenge by hashing the 16-byte challenge received in step (2), the 16-byte Peer Authenticator Challenge generated in step (3a), and the Client's username.
3c. Client creates a 24-byte reply, using the Windows NT hash function and the 8-byte challenge generated in step (3b). This process is identical to MS-CHAPv1.
3d. The Client sends the Server the results of steps (3a) and (3c).
MS-CHAP v2

4a. The Server uses the hashes of the Client's password, stored in a database, to decrypt the replies. If the decrypted blocks match the challenge, the Client is authenticated.

4b. The Server uses the 16-byte Peer Authenticator Challenge from the client, as well as the Client's hashed password, to create a 20-byte Authenticator Response."

5. The Client also computes the Authenticator Response. If the computed response matches the received response, the Server is authenticated.
Deriving the 20-byte Authenticator Response

- The Server (or the Client) hashes the 16-byte NT password hash with to get password-hash-hash. (The Server stores the client's password hashed with MD4; this is the NT password hash value.)
- The Server concatenates the password-hash-hash, the 24-byte NT response, and the literal string "Magic server to client constant", and then hashes the result with SHA.
- The Server concatenates the 20-byte SHA output from step (2), the initial 8-byte generated challenge and the literal string "Pad to make it do more than one iteration", and then hashes the result with SHA.
- Resulting 20 bytes are the authenticator response.
Deriving MPPE keys from MS-CHAPv2 credentials

- Deriving MPPE keys from MS-CHAPv2 credentials works as follows:
  - Hash 16-byte NT password hash, 24-byte response from MS-CHAPv2 exchange, and a 27-byte constant (the string "This is the MPPE Master Key") with SHA. Truncate to get 16-byte master-master key.
  - Using a deterministic process, convert the master-master key to a pair of session keys. For 40-bit session keys, this is done as follows:
    - Hash the master-master key, 40 bytes of 0x00, an 84-byte constant and 40 bytes of 0xF2 with SHA. Truncate to get an 8-byte output.
    - Set the high-order 24 bits of 0xd1269e, resulting in a 40-bit key.
    - The magic constants are different, for keys to encrypt traffic from the Client to the Server and from Server to the Client.
  - For 128-bit session keys, the process is as follows:
    - Hash master-master key, 40 bytes of 0x00, 84-byte constant (magic constant 2 or 3), and 40 bytes of 0xF2 with SHA. Truncate to 16-byte output.
Version Rollback Attack

- Since Microsoft has attempted to retain some backwards compatibility with MS-CHAPv1, it is possible for an attacker to mount a "version rollback attack".
- In this attack, the attacker convinces both the Client and the Server not to negotiate the more secure MS-CHAPv2 protocol, but to use the less secure MS-CHAPv1 protocol.
- Microsoft claims that the operating systems will try to negotiate MS-CHAPv2 first, and only drop back to MS-CHAPv1 if the first negotiation fails.
- Additionally, it is possible to set the Server to require MS-CHAPv2.
  - Software switches to turn off backwards compatibility are registry settings, and can be difficult to find.
  - Since older versions of Windows cannot support MS-CHAPv2, backwards compatibility must be turned on if there are any legacy users on the network.
- Hence version rollback attacks are a significant threat.
### MS-CHAP v1 and v2

<table>
<thead>
<tr>
<th>MS-CHAP Version 1</th>
<th>MS-CHAP Version 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Negotiates CHAP with an algorithm value of 0x80.</td>
<td>Negotiates CHAP with an algorithm value of 0x81.</td>
</tr>
<tr>
<td>Server sends an 8-byte challenge value.</td>
<td>Server sends a 16-byte value to be used by the client in creating an 8-byte challenge value.</td>
</tr>
<tr>
<td>Client sends 24-byte LANMAN and 24-byte NT response to 8-byte challenge.</td>
<td>Client sends 16-byte peer challenge that was used in creating the hidden 8-byte challenge, and the 24-byte NT response.</td>
</tr>
</tbody>
</table>
**MS-CHAP v1 and v2**

<table>
<thead>
<tr>
<th>Client decides to continue or end based upon the SUCCESS or FAILURE response above.</th>
<th>Server sends a response stating SUCCESS or FAILURE and piggybacks an Authenticator Response to the 16-byte peer challenge.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server sends a response stating SUCCESS or FAILURE.</td>
<td>Client decides to continue or end based upon the SUCCESS or FAILURE response above. In addition, Client checks the validity of the Authenticator Response and disconnects if it is not the expected value.</td>
</tr>
</tbody>
</table>
Further Reading