CS 6324: Information Security
Network Security Protocols & Attack Strategies

Dr. Junia Valente

Department of Computer Science
The University of Texas at Dallas

(contains material from: “Security Engineering” by Ross Anderson, and Dr. Alvaro Cardenas’ Info Sec slides)
Announcement:

- Posted homework #1 on eLearning (due on February 23 @ 10 a.m.)
Outline

✧ **SECURITY PROTOCOLS**

✧ SSL

✧ IPSec

✧ Network Attacks
Outline

✧ SECURITY PROTOCOLS
  ✧ SSL
  ✧ IPSec
✧ Network Attacks
SSL:

- SSL v2 deployed in Netscape Communicator 1.1 (1995)
- SSL v3 standardized as TLS

- Major success story!
  of applying crypto in the real world!

- The “lock” in your browser
SSL: Certificates

One of the hundreds of certificate authorities (CAs) trusted by your browser is binding the public key your browser will use to the domain name you are accessing.

e.g., the public key that your browser will use to encrypt the communication belongs to the domain you are accessing.

proof there is a binding
**What a security certificate is:**

“When you go to a site that uses HTTPS (connection security), the website's server uses a certificate to prove the website's identity to browsers, like Chrome. Anyone can create a certificate claiming to be whatever website they want.

To help you stay on safe on the web, Chrome requires websites to use certificates from trusted organizations.”

~Google Chrome Help
SSL: Certificates

Lock may not be enough
SSL: Certificates

Malaysian Police Website Found To Be Hosting A Phishing Page

“... SSL certificates offer very limited protection from phishing.”

“A PayPal phishing page has been discovered on the official portal of the Malaysian police force for Johor region. It was used to trick the users into surrendering their login information, despite the website having a valid SSL certificate.”

Trust issues: “The illusion of safety is created by an SSL certificate which is unconditionally accepted by several major browsers, including Firefox and Safari.”
SSL:
To mitigate these problems, we have Extended Validation Certificates

- Better verification of identity
- More visual cues to the user
  - e.g., name of company – not just domain name – in green
- Fewer CAs authorized to issue them
SSL: 
Extended Validation Certificates in different browsers
Extended Validation Certificates are Dead

That's it - I'm calling it - extended validation certificates are dead. Sure, you can still buy them (and there are companies out there that would just love to sell them to you!), but their usefulness has now descended from "barely there" to "as good as non-existent". This change has come via a combination of factors including increasing use of mobile devices, removal of the EV visual indicator by browser vendors and as of today, removal from Safari on iOS (it'll also be gone in Mac OS Mojave when it lands next week):
Certificate Pinning:

- How many certificate authorities (CAs) do you trust?

- Browser has a list of hundreds of them approved

- In a few instances, attackers have broken into a certificate company (e.g., DigiNotar, Comodo)
  And they can generate a valid certificate for any website they want!
  Do MiTM attacks!

Certificate pinning solves this problem:

- Browser only accepts one CA for a given hostname
Outline

✧ SECURITY PROTOCOLS

✧ SSL

✧ IPSec

✧ Network Attacks
IPSec:

- Security built into the IP layer

- Designed as part of the IPv6 standard

- Can be used as an extension to IPv4

- Provides support to Virtual Private Networks (VPNs)

- Provides support for authentication and confidentiality

- Defines infrastructure for key management
  - e.g., establishes a mutual authentication between two parties (at the beginning of a session) and negotiation of cryptographic keys to use during the session.
- Extends a private network across a public network, as if a user’s device was directly connected to the private network.
Connectivity Deployed using the Public Internet with the Same Security and ‘Performance’ as a Private Network
Cisco AnyConnect VPN

The Cisco AnyConnect VPN allows users to remotely connect to the campus network in order to access on-premise resources. In order to utilize the UTD Cisco AnyConnect VPN, please either enroll your mobile device or request a hardware token with NetIDplus at the NetID management site. Additional information regarding NetIDplus can be found here.

Once connected to the VPN, you will be able to:

- Access campus network drives. (H, P, G, etc.)
- Work on your office computer with Remote Desktop.
- Use site-licensed software like Matlab and Mathematica.

**Note:** The UTD VPN is not configured to work from inside the UTD network. The Cisco AnyConnect Client software is only supported for use off-campus.

- Enroll in NetIDplus
- Install the Cisco AnyConnect VPN software for your Windows or Mac computer:
  - Install Cisco VPN – Windows
  - Install Cisco VPN – Mac

IPSec:
Using VPN @ UTD
IPSec: Using VPN @ UTD: two-factor authentication
IPSec: Virtual Private Networks (VPNs)

[Diagram showing a secure tunnel between a computer and a server, indicating that a hacker cannot steal your data and your ISP cannot spy on you.]

https://privacycanada.net/online-privacy-guide/
- Security association (SA)
  - A security association is defined before starting to communicate
  - The security association specifies the security parameters for a particular communication channel:
    - Algorithms used for authentication
    - Algorithms used for bulk encryption
    - Validity of the association

Two options/parts:
- Authentication Header (AH)
- Encapsulating Security Payload (ESP)
IPSec: Basic Concepts

Two options/parts:

- Authentication Header (AH)
  - Extension that provides authentication and integrity with HMAC
  - Make sure packet header is not changed in transit

- Encapsulating Security Payload (ESP)
  - Extension that provides confidentiality
  - Encrypts the payload

- If wanted to have confidentiality and integrity, we’d use both AH and ESP; now, ESP was extended, so you can use ESP for authentication, integrity, and confidentiality.
Outline

✧ Security Protocols

✧ SSL

✧ IPSec

✧ NETWORK ATTACKS
Network Attacks:

- **Spoofing** – impersonation of a host

- Denial of service (DoS)

- Botnets, worms, viruses, trojans, etc.

- **Sniffing** – access to information
  - Using a network sniffer device
  - or, putting our computer in promiscuous mode
Attack Strategies:

Traditional model:
- Identify vulnerable device:
  scanning, network enumeration, OS fingerprinting, etc.

New model:
- Includes the traditional model and more:
  - “Let vulnerable computers find the attacker”
  - Attacker may put a compromised web server, and attack all browsers when they try connecting to it.
Network Sniffing:

- Technique at the basis of many attacks

- The attacker sets his/her network interface in promiscuous mode

- Can access all the traffic on the segment
Sniffers:

- Sniffers are typically passive programs

- They put the **network interface** in promiscuous mode

- **ifconfig** (popular network interface configuration utility in Mac and Linux)
  - e.g., we can use to set the IP address of a network interface and disabling/enabling an interface

```
eth0  Link encap:Ethernet HWaddr 00:10:4B:E2:F6:4C
     inet addr:192.168.1.20 Bcast:192.168.1.255 Mask:255.255.255.0
     UP BROADCAST RUNNING PROMISC MULTICAST MTU:1500 Metric:1
     RX packets:1016 errors:0 dropped:0 overruns:0 frame:0
     TX packets:209 errors:0 dropped:0 overruns:0 carrier:0
     collisions:0 txqueuelen:100
```
- **tcpdump** is a tool that analyzes the traffic on a network segment

- One of the most used / most useful tools

- Based on libpcap, which provides a platform-independent library and API to perform traffic sniffing

- Allows one to specify an expression that defines which packets have to be captured

- Requires **root** privileges to be able to set the interface in promiscuous mode (privileges not needed when reading from file)
Wireshark: Most popular sniffer

Example of traffic sent in plain-text and using PKCS#7 to sign and/or encrypt messages under a PKI.

We can see the communication between devices.
Wireshark: Most popular sniffer
- Used to determine which UDP services are available

- A zero-length UDP packet is sent to each port

  - If an Internet Control Message Protocol (ICMP) error message “port unreachable” is received, then the service is assumed to be unavailable

  - Many TCP/IP stack implementations implement a limit on the error message rate, therefore this type of scan can be slow (e.g., Linux limits 80 messages every 4 seconds)
Port Scan:

- **nmap** most popular port scanner

- Example scanning UDP ports on a device with IP address 192.168.1.10

```
% nmap -sU 192.168.1.10
Starting nmap by fyodor@insecure.org ( www.insecure.org/nmap/ )
Interesting ports on (192.168.1.10):
(The 1445 ports scanned but not shown below are in state: closed)
  Port         State  Service
137/udp       open    netbios-ns
138/udp       open    netbios-dgm
Nmap run completed -- 1 IP address (1 host up) scanned in 4 seconds
```
Port Scan: Scanning TCP ports

- Used to determine TCP services available on a victim host

- Most services are statically associated with port numbers
  e.g., see `/etc/services` in UNIX systems

- In its simplest form (e.g., `connect()` scanning), the attacker tries to open a TCP connection to all the 65535 ports of the victim host

- If the handshake is successful then the service is available

- Advantage: no need to be root
- Disadvantage (?): very noisy! (Be careful!)
- Example scanning TCP ports on a device with IP address 192.168.1.20

```
# nmap -sT 192.168.1.20
Starting nmap by fyodor@insecure.org ( www.insecure.org/nmap/ )
Interesting ports on (192.168.1.20):
(The 1500 ports scanned but not shown below are in state: closed)
<table>
<thead>
<tr>
<th>Port</th>
<th>State</th>
<th>Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>7/tcp</td>
<td>open</td>
<td>echo</td>
</tr>
<tr>
<td>9/tcp</td>
<td>open</td>
<td>discard</td>
</tr>
<tr>
<td>11/tcp</td>
<td>open</td>
<td>systat</td>
</tr>
<tr>
<td>13/tcp</td>
<td>open</td>
<td>daytime</td>
</tr>
<tr>
<td>15/tcp</td>
<td>open</td>
<td>netstat</td>
</tr>
<tr>
<td>19/tcp</td>
<td>open</td>
<td>chargen</td>
</tr>
<tr>
<td>21/tcp</td>
<td>open</td>
<td>ftp</td>
</tr>
<tr>
<td>22/tcp</td>
<td>open</td>
<td>ssh</td>
</tr>
<tr>
<td>23/tcp</td>
<td>open</td>
<td>telnet</td>
</tr>
<tr>
<td>512/tcp</td>
<td>open</td>
<td>exec</td>
</tr>
<tr>
<td>513/tcp</td>
<td>open</td>
<td>login</td>
</tr>
<tr>
<td>514/tcp</td>
<td>open</td>
<td>shell</td>
</tr>
<tr>
<td>6000/tcp</td>
<td>open</td>
<td>X11</td>
</tr>
</tbody>
</table>

Nmap run completed – 1 IP address (1 host up) scanned in 0 seconds
```
Other usage of scanning

- **OS fingerprinting** allows one to determine the operating system of a host by examining the reaction to carefully crafted packets.

- Also for **discovering hosts** that are “up” in a network.