

- Security in Networks
 - Threats in Networks

Intelligence

- Social Engineering
 - Attacker gathers sensitive information directly from a person
 - Often, the attacker pretends to be somebody within the person's organization
 - who has a problem and exploits the person's willingness to help (or vice versa)
 - I forgot my password, I locked myself out, there's a problem with your Paypal account,...
- Dumpster diving
- Eavesdropping on oral communication between people
- Google
 - There's lots of information on the Internet that shouldn't be there
 - The right Google query will find it

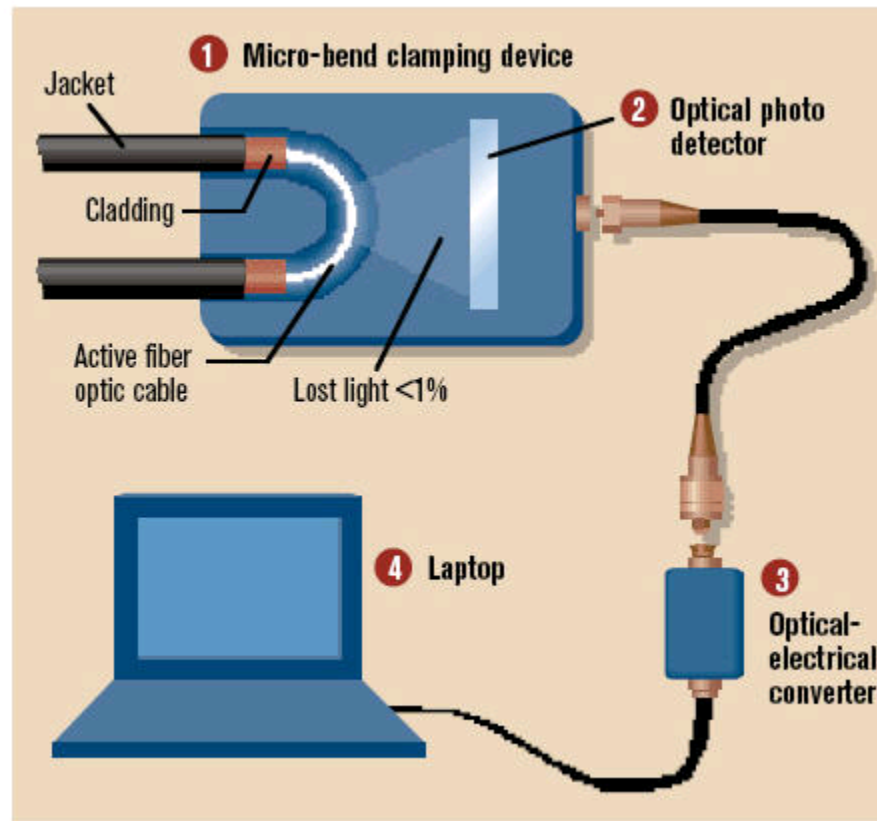
Eavesdropping and Wiretapping

- Owner of node can always monitor communication flowing through node
 - Eavesdropping or passive wiretapping
 - Active wiretapping involves modification or fabrication of communication
- Can also eavesdrop while communication is flowing across a link
 - Degree of vulnerability depends on type of communication medium
- Or when communication is accidentally sent to attacker's node
- It is prudent to **assume that your communication is wiretapped**

Communication Media

- Copper cable
 - Inductance allows a physically close attacker to eavesdrop without making physical contact
 - Cutting cable and splicing in secondary cable is another option
- Optical fiber
 - No inductance, and signal loss by splicing is likely detectable
 - However, just bending the fiber might work
- Microwave/satellite communication
 - Signal path at receiver tends to be wide, so attacker close to receiver can eavesdrop
- All these attacks are feasible in practice, but require **physical expenses/effort**

Fiber Tapping



(Sandra Kay Miller, Information Security Magazine, November 2006)

See also http://www.schneier.com/blog/archives/2007/09/eavesdropping_o_1.html

Communication Media (cont.)

- WiFi
 - Can be **easily intercepted** by anyone with a Wi-Fi-capable (mobile) device
 - Don't need additional hardware, which would cause suspicion
 - Maybe from kilometers away using a directed antenna
 - WiFi also raises other security problems
 - Physical barriers (walls) help against random devices being connected to a wired network, but are (nearly) useless in case of wireless network
 - Need authentication mechanism to defend against free riders

Misdelivered Information

- Local Area Network (LAN)
 - Connects all computers within a company or a university
 - Technical reasons might cause a packet to be sent to multiple nodes, not only to the intended receiver
 - By default, a network card ignores wrongly delivered packets
 - An attacker can change this and use a **packet sniffer** to capture these packets
- Email
 - Wrongly addressed emails, inadvertent Reply-To-All

Impersonation

- Impersonate a person by stealing his/her password
 - Guessing attack
 - Exploit default passwords that have not been changed
 - Sniff password (or information about it) while it is being transmitted between two nodes
 - Social engineering
- Exploit trust relationships between machines/accounts
 - Rhosts/rlogin mechanism allows user A on machine X to specify that user B on machine Y can act as A on X without having to re-enter password
 - shosts/slogin mechanism is similar
 - Attacker breaking into machine Y can exploit this
 - Or attacker might be able to masquerade as machine Y

Spoofing

- An object (node, person, URL, Web page, email, WiFi access point,...) masquerades as another one
- URL spoofing
 - Exploit typos: www.uwaterlo.ca
 - Exploit ambiguities: www.foobar.com or www.foo-bar.com?
 - Exploit similarities: www.paypa1.com
- Web page spoofing and URL spoofing are used in Phishing attacks
- “Evil Twin” attack for WiFi access points
- Spoofing is also used in session hijacking and man-in-the-middle attacks

Session Hijacking

- TCP protocol sets up state at sender and receiver end nodes and uses this state while exchanging packets
 - e.g., sequence numbers for detecting lost packets
 - Attacker can hijack such a session and masquerade as one of the endpoints
- Web servers sometimes have client keep a little piece of data (“cookie”) to re-identify client for future visits
 - Attacker can sniff or steal cookie and masquerade as client
- Man-in-the-middle attacks are similar; attacker becomes stealth intermediate node, not end node

Traffic Flow Analysis

- Sometimes, the mere existence of communication between two parties is sensitive and should be hidden
 - Whistleblower
 - Military environments
 - Two CEOs
- TCP/IP has each packet include unique addresses for the packet's sender and receiver end nodes
- Attacker can learn these by sniffing packets
- More on protecting yourself from this attack later

Integrity Attacks

- Attacker can modify packets while they are being transmitted
 - Change payload of packet
 - Change address of sender or receiver end node
 - Replay previously seen packets
 - Delete or create packets
- Line noise, network congestion, or software errors could also cause these problems
 - TCP/IP will likely detect environmental problems, but fail in the presence of an active attacker
 - How in the case of TCP's checksumming mechanism?

Integrity Attacks (cont.)

- DNS cache poisoning
 - Domain Name System maps hostnames (www.uwaterloo.ca) to numerical addresses (129.97.128.40), as stored in packets
 - Attacker can create wrong mappings

Protocol Failures

- TCP/IP assumes that all nodes implement protocols faithfully
- E.g., TCP includes a mechanism that asks a sender node to slow down if the network is congested
 - An attacker could just ignore these requests
- Some implementations do not check whether a packet is well formatted
 - E.g., the value in the packet's length field could be smaller than the packet's actual length, making buffer overflow possible
 - Potentially disastrous if all implementations are from the same vendor or based on the same code base
- Protocols can be very complex, behavior in rare cases might not be (uniquely) defined
- Some protocols include broken security mechanisms
 - WEP (see later)

Web Site Vulnerabilities

- Accessing a URL has a web server return HTML code
 - Tells browser how to display web page and how to interact with web server
 - Attacker can examine this code and find vulnerabilities
- Web site defacements
- Attacker crafts malicious URL and sends it to web server
 - to exploit a buffer overflow
 - to invoke a shell or some other program
 - to feed malicious input parameters to a server-side script
 - to access sensitive files
 - E.g., by including “../” in a URL or by composing URLs different from the “allowed ones” in the HTML code

Web Site Vulnerabilities (cont.)

- HTTP protocol is stateless, so web server asks client to keep state when returning a web page and to submit this state when accessing next web page
 - Cookie or URL
(<http://www.store.com?clientId=4342>)
 - Attacker can submit **modified** state information
- Cross-site scripting (XSS) attacks
 - Attacker adds his/her own HTML code to somebody else's web page
 - E.g., in the comments section of a blog
 - Code could contain a virus
 - Other users download and execute this code when downloading the web page

Denial of Service (DoS)

- Cutting a wire or jamming a wireless signal
- Flooding a node by overloading its Internet connection or its processing capacity
- Ping flood
 - Node receiving a ping packet is expected to generate a reply
 - Attacker could overload victim
 - Different from “ping of death”, which is a malformed ping packet that crashes victim’s computer
- Smurf attack
 - Spoof address of sender end node in ping packet by setting it to victim’s address
 - Broadcast ping packet to all nodes in a LAN

Denial of Service (cont.)

- Exploit knowledge of implementation details about a node to make node perform poorly
- SYN flood
 - TCP initializes state by having the two end nodes exchange three packets (SYN, SYN-ACK, ACK)
 - Server queues SYN from client and removes it when corresponding ACK is received
 - Attacker sends many SYNs, but no ACKs
- Send packet fragments that cannot be reassembled properly
- Craft packets such that they are all hashed into the same bucket in a hash table

Denial of Service (cont.)

- Black hole attack
 - Routing of packets in the Internet is based on a distributed protocol
 - Each router informs other routers of its cost to reach a set of destinations
 - Malicious router announces low cost for victim destination and discards any traffic destined for victim
 - Has also happened because of router misconfiguration
- DNS attacks
 - DNS cache poisoning can lead to packets being routed to the wrong host

Distributed Denial of Service (DDoS)

- If there is only a single attacking machine, it might be possible to identify the machine and to have routers discard its traffic (see later)
- Difficult if there are lots of attacking machines
- Most might participate without knowledge of their owners
 - Attacker breaks into machines using Trojan, buffer overflow,... and installs malicious software
 - Machine becomes a **zombie/bot** and waits for attack command from attacker
 - A network of bots is called a **botnet**
 - How would you turn off a botnet?

Botnets

- Today's botnets are very sophisticated and include
 - Virus/worm/trojan for propagation based on multiple exploits
 - Stealthiness to hide from owner of computer
 - Code morphing to make detection difficult
 - Bot usable for different attacks (spam, DDoS,...)
 - Distributed, dynamic & redundant control infrastructure
- Earlier worms (Nimda, slammer) were written by hackers for **fame** with the goal to spread worm as fast as possible
 - slammer infected 75,000 hosts in 10 minutes
 - Caused disruption and helped detection

Botnets (cont.)

- Botnets are controlled by hackers looking for **profit**, which rent them out
 - Criminal organizations
- Spread more slowly, infected machine might lie dormant for weeks
- Currently, **Storm Worm** botnet is expected to include millions of machines, its processing power likely makes it the world's biggest supercomputer
- We don't know when and for what it will be used and who is behind it

Active Code

- To reduce load on server, server might ask client to execute code on its behalf
 - Java, JavaScript, ActiveX
 - Invoke another application (Word, iTunes,...)
 - Maybe inadvertently (see XSS attack)
- Obviously, this can be dangerous for client
- Java 1.1 ran in a sandbox with limited capabilities, code is checked for correctness
 - No writing to a file, no talking to random network nodes
 - Similar for JavaScript
 - But it could still use up CPU or memory resources, wreak havoc with display, or play annoying music

Active Code (cont.)

- Java 1.2 can break out of sandbox if approved by user
 - What's the problem here?
- ActiveX
 - No sandbox or correctness check
 - Downloaded code is cryptographically signed, signature is verified to be from “trusted” entity before execution
- Third-party applications
 - Turn out to be a huge problem, for all browsers
 - Malicious input parameters, Word macros,...
 - Potentially disastrous if application has full access rights to a user's account

Recap

- Security in Networks
 - Threats in Networks

Next time

- Security in Networks
 - Network Security Controls
 - Firewalls
 - Honeypots
 - Intrusion Detection Systems