Lecture Notes on Jan/30/2023

Chapter 2:
Network Communications Protocols and Built-in Security (Part 02)

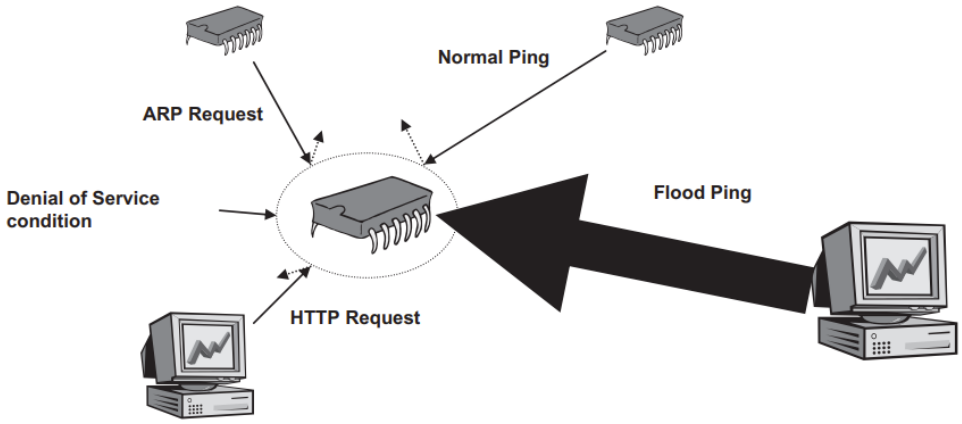
CYENG 351: Embedded Secure Networking

Instructor: Dr. Shayan (Sean) Taheri

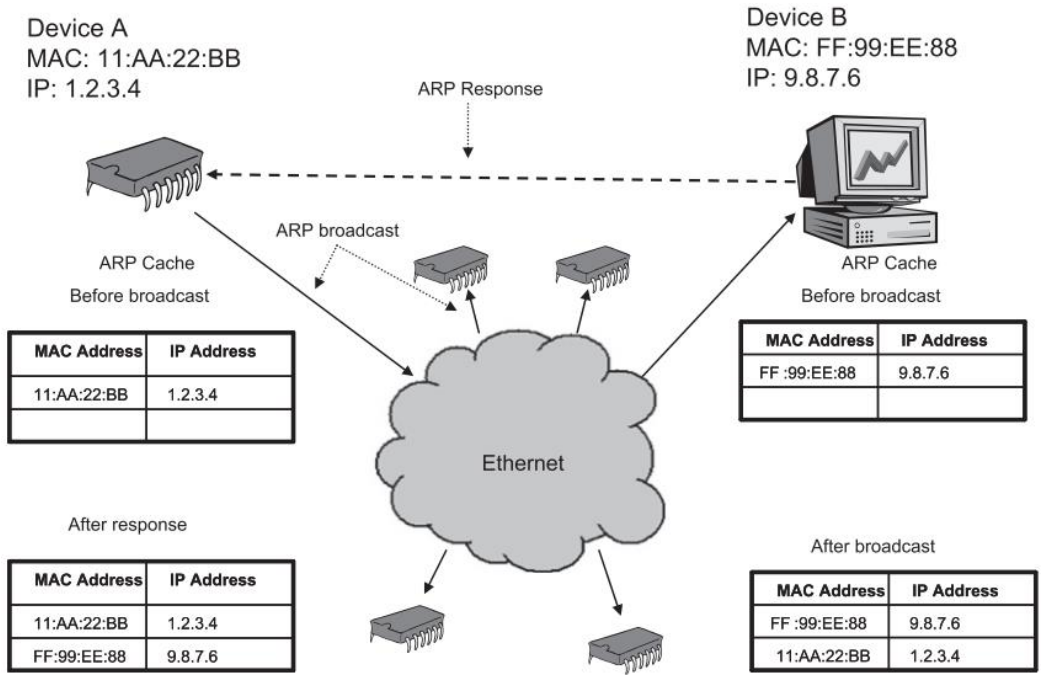
Gannon University (GU)

Ethernet and ARP (Cont.)

- Once the target device receives the message, which contains the **sender's hardware address**, it updates **its own ARP table** and sends a reply back; this causes the sender's table to be updated.
- In this fashion, the ARP table does not have to be constantly updated, sending flurries of packets over the network whenever a connection is desired.
- The sender and receiver can contact directly using the **stored hardware addresses**.
- The trick for preventing DOS attacks on our hypothetical device involves the central machine and the device each communicating a periodic "**ping**," where each side issues a request and expects a reply.
- All in all, ARP is very important as it forms the basis for **dynamic Ethernet networks**, but it is, unfortunately, dangerous due to its inherent security pitfalls.



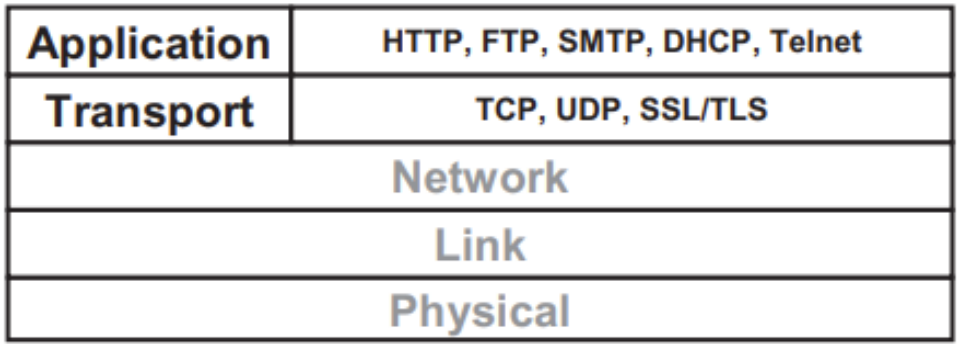
Flood Ping Can Bring a Device to a Grinding Halt



ARP Broadcast Steps

Transport and Internet Layer Protocols

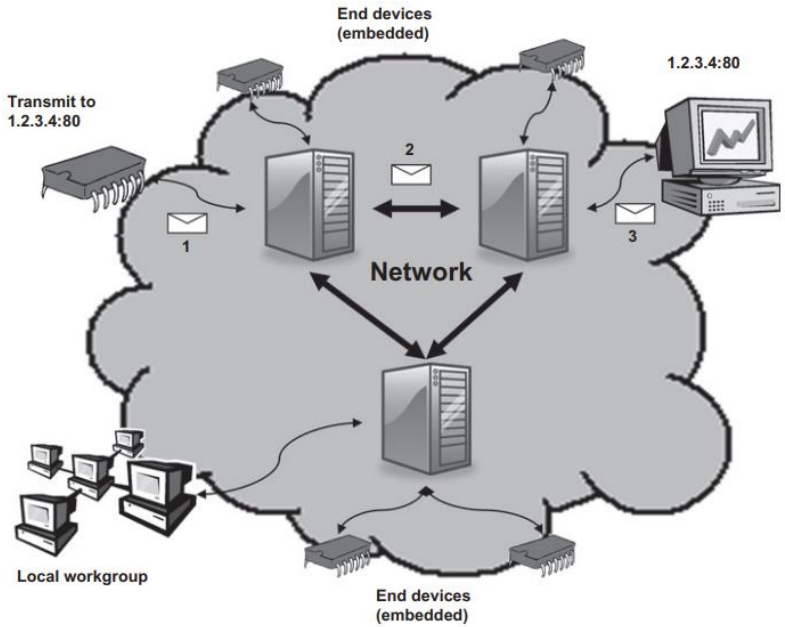
- The transport layer is the most common lower layer that an application designer will deal with.
- The lowest level protocols that we have already covered are usually buried deep within hardware-specific driver code, or in hardware itself.
- It is in the transport layer that the application engineer first has significant control over the network settings and how his or her application will interact with the network stack.
- The **Transport Control Protocol (TCP)** and the **Internet Protocol (IP)**, form the software basis of the Internet, and their use is so widespread that it would be very rare to find a network-capable application that did not support them.
- The **User Datagram Protocol (UDP)**, which can be thought of as "**TCP-lite**," since it works on the same basic principles as TCP but does not have the reliability guarantees provided by TCP.



Transport and Application Layers in Network Stack

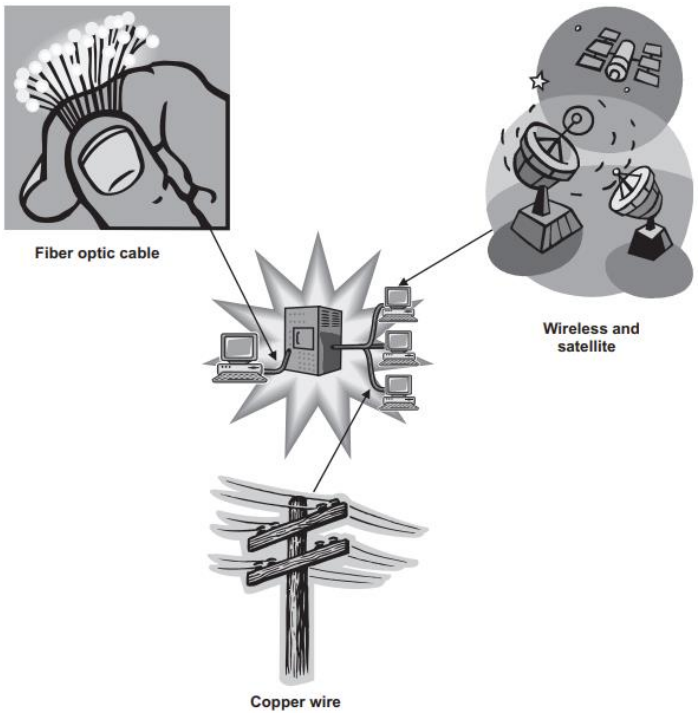
Transport Control Protocol and the Internet Protocol (TCP/IP)

- TCP was one of the earliest "**reliable**" protocols, in that the application does not have to worry about the fragmentation of information.
- To the application, TCP is like a pipe (or a queue) in that what goes in one end will always come out the other end in the same order it entered.
- TCP utilizes several methods to keep data in order and to assure all data is sent and received, and as we will see, this property is extremely important to the security of a networked application.
- TCP in and of itself does not provide any encryption or explicit security features, but the reliability and robustness of the protocol make it the perfect foundation for a transport layer security protocol.
- TCP provides reliability based on the idea of **a dynamically-sizable "window"** that the protocol can adjust to compensate for network traffic or other problems.



TCP/IP Packets in the Network

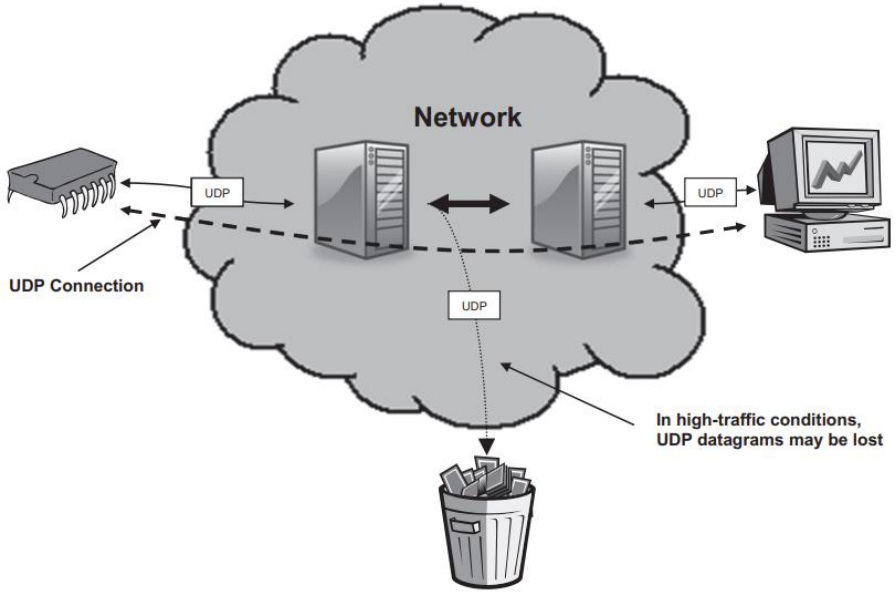
Different Media Types Used for Communication



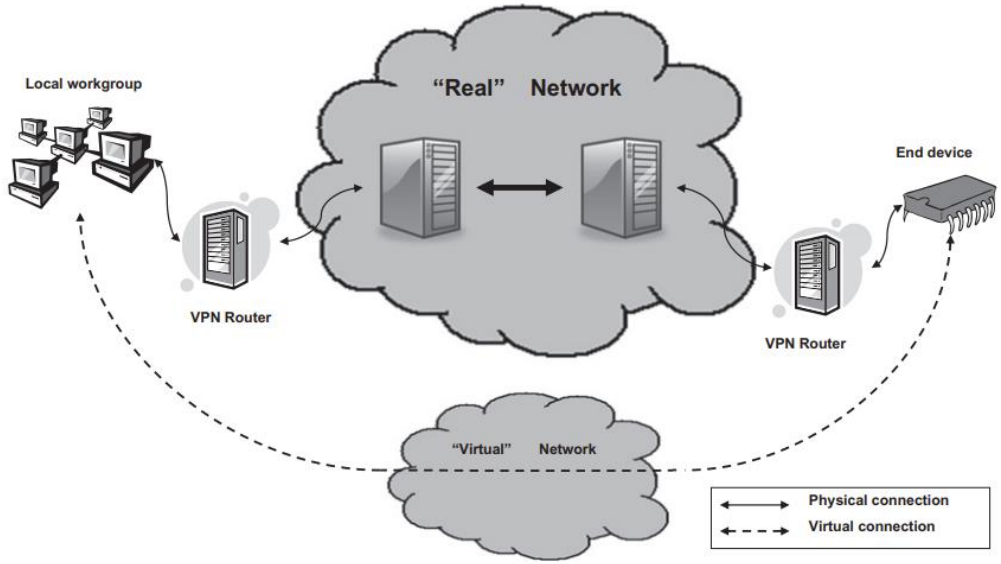
User Datagram Protocol (UDP)

- TCP is a relatively heavyweight protocol due to the extensive logic required to provide reliability.
- The User Datagram Protocol fills this niche, since it is a simple protocol without any of the reliability guarantees of TCP.
- UDP packets, called **datagrams**, are simply sent to a target address with essentially no guarantee that any will arrive.
- The reason UDP works at all is that the underlying network (usually consisting of IP over some low-level protocol) is for the most part reliable.
- Applications using UDP must be tolerant of dropped or out-of-order data grams, as the protocol does not catch these errors.
- The inherent lack of reliability in UDP makes it unsuitable for most secure applications, but it does have a useful property that TCP does not—multicasting.

UDP Is Unreliable—Datagrams May Be Lost in Transit

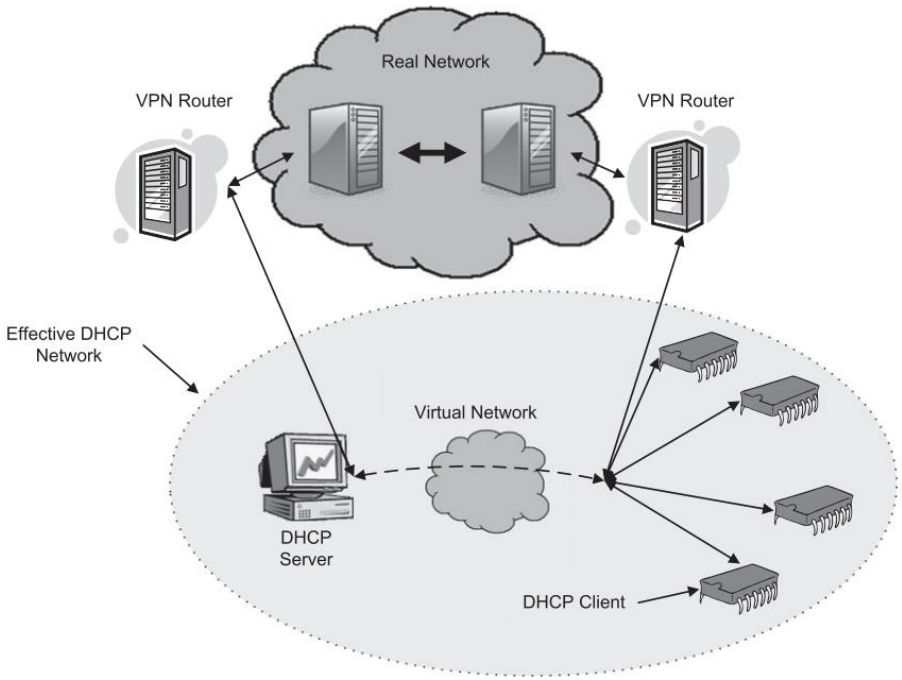


Virtual Private Network



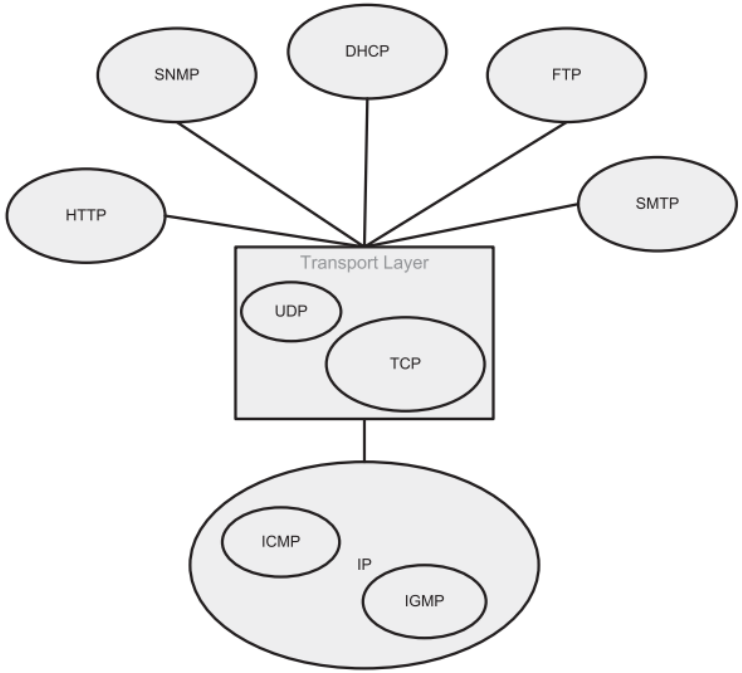
Other Network Protocols

- **Other protocols** are the Dynamic Host Configuration Protocol (DHCP), the Simple Network Management Protocol (SNMP), and subprotocols like the Internet Control Message Protocol (ICMP) and the Internet Group Management Protocol (IGMP), which are components of the IP protocol suite.
- There are a variety of other protocols that you may be familiar with, such as the Simple Mail Transfer Protocol (SMTP, or email), File Transfer Protocol (FTP), or the Hypertext Transfer Protocol (HTTP).
- **DHCP** allows devices to connect to a network without having been previously configured for that particular network.
- Once the low-level protocol establishes a connection (physical level, such as Ethernet), a DHCP server provides all the IP configuration information for that network to the connecting device.
- If DHCP is required, the application should utilize a higher-level mechanism to authenticate new clients, and the network should be designed to authenticate new DHCP servers in some way.



DHCP Used in a VPN

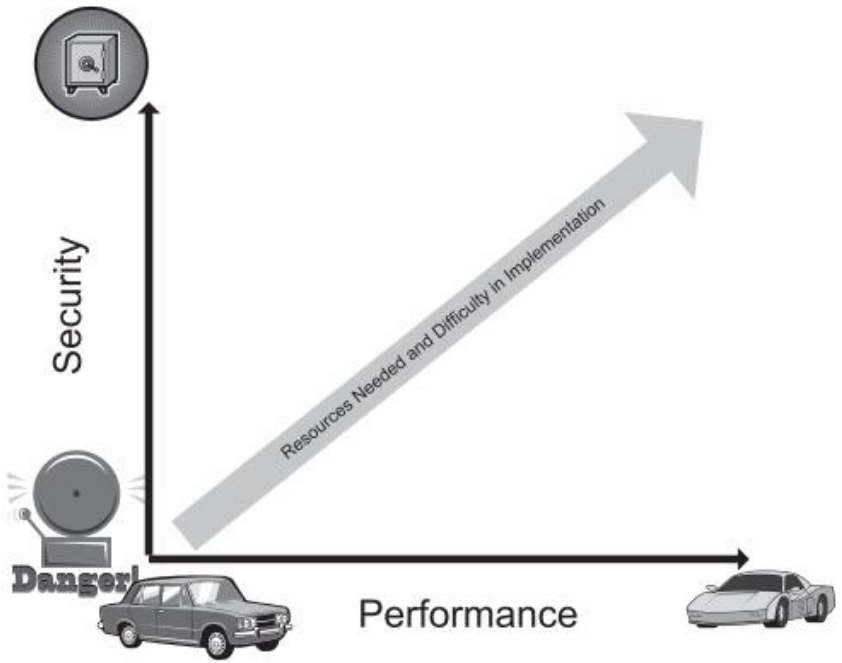
IP Protocol Suite



Why All the Protocol Analysis?

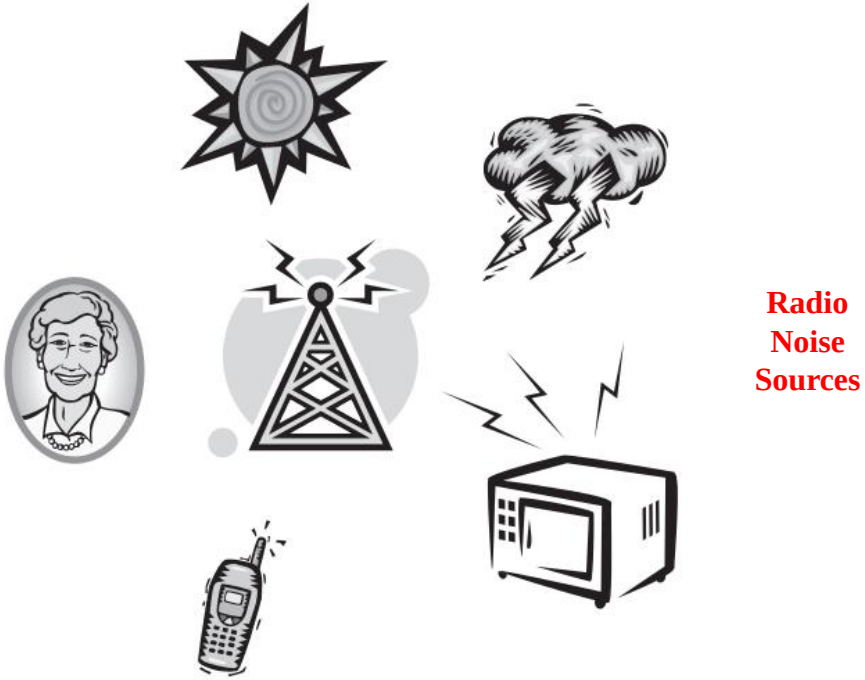
- Security is an add-on feature, and as such, cannot simply be ignored.
- It would be nice if all the networking protocols included some reliable security, or (even better) were inherently secure in their designs.
- The problem is that many of the protocols were designed for performance, and security is almost always at odds with high performance.
- As a result, security is forgotten, and it is up to the end engineer to tack on the security at the last minute.
- **Security is not something that just "happens" to an application but is something that must be deliberately designed and carefully integrated into all parts of an application, starting at the lowest protocol levels**.
- If you need security, you need to be willing to look at parts of the system that others would be quite willing to ignore (certainly the hackers won't ignore those parts).

Security vs. Performance and Ease of Implementation



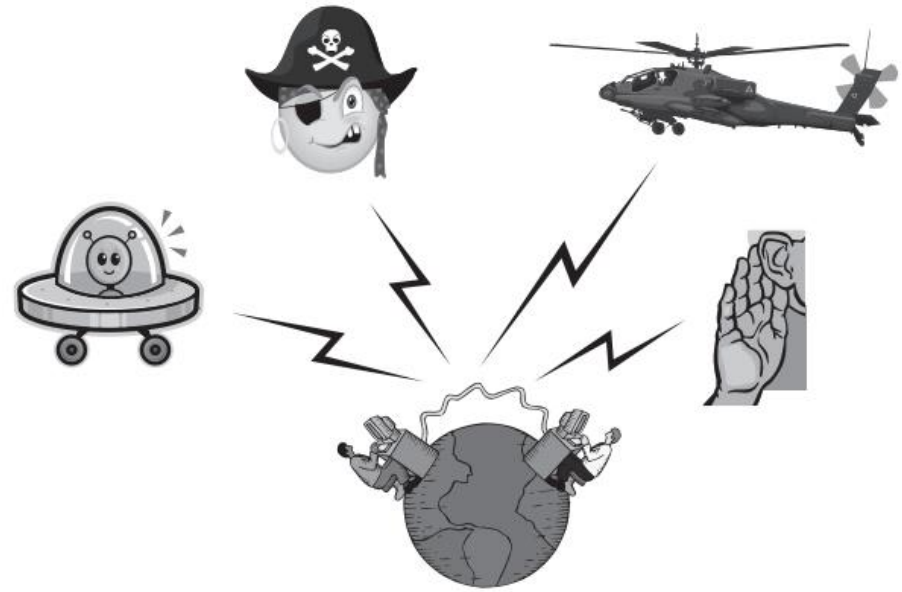
Cutting the Wires

- The wired protocols, unless running over a public network, do not really need all that much security, except in the most critical applications.
- These protocols all rely on an inherent property of the network itself for basic security— wires are easy to secure.
- This property does not become apparent until you try to remove the wires and broadcast information using a radio.
- If someone were to try to eavesdrop on a wire, they would need physical access to that wire.
- Without the wire, all they need is an antenna (and maybe a dish to amplify the signal).
- One of the primary reasons **wireless technologies** have not been as prevalent so far (even though the radio predates the Internet) is that wireless communication is hard—everything from cell phones to sunspots to microwave ovens to Grandma's pacemaker emits some kind of radio noise.
- Security is basically an absolute necessity for any type of wireless communication technology.



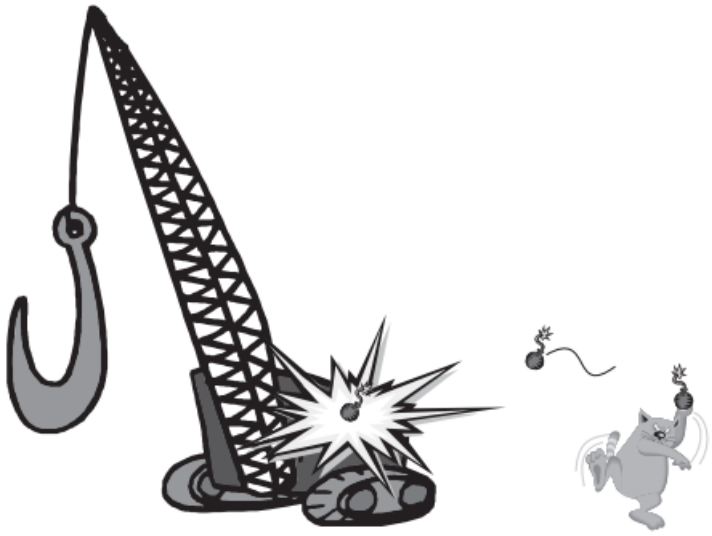
Radio Noise Sources

Wireless Communication Threats



Wireless Technologies

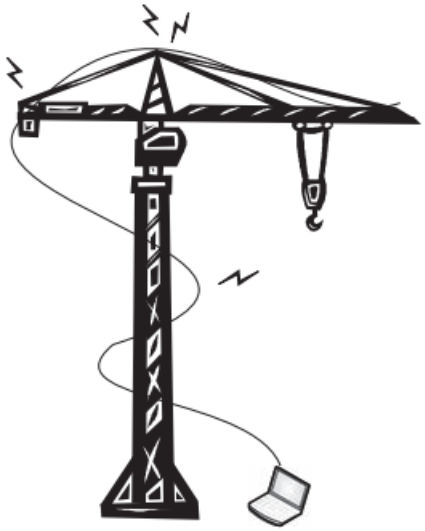
- With the dawning of a wireless revolution, there will be a similar revolution for embedded technology, as wireless capability will allow for applications that never were possible before.
- The current standard leading the way is actually **a collection of wireless networking protocols**, similar and related to Ethernet technology.
- By far one of the most mature wireless standards, Wi-Fi (802.11a/b/g and other variants) is the dominant standard for computer systems.
- Cellular technology is bigger, but less Internet-friendly, and generally used for telephone communications.
- A modern computer or embedded device requires a reliable, robust communications medium, since even a single bit off can cause major problems for an application.
- The Wi-Fi protocol, as defined in the IEEE 802.11 standard, is designed to **mimic the properties of a standard "wired" protocol**.
- **ZigBee** is a very flexible wireless protocol and suitable for embedded applications, allowing for little or no configuration of individual devices when establishing a network.



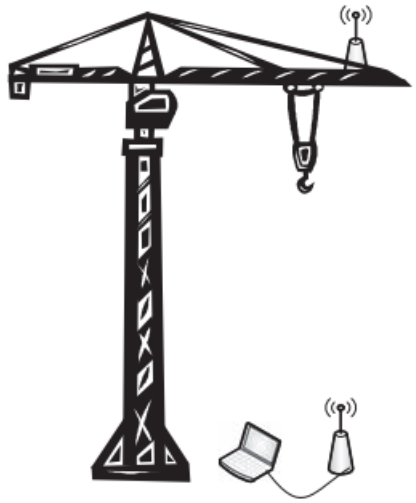
Eavesdropping may not be your biggest concern...

A Really Bad Day at Work . . .

Wired vs. Wireless Crane Application



Wires are a Liability



Going wireless simplifies the application

Assignment

- **Reading Assignment:**
  - Stapko, T., 2011. **Practical embedded security: building secure resource-constrained systems**. Elsevier.
    - ✓ "Chapter 2: Network Communications Protocols and Built-in Security", Pages 28-48.

Questions?