Lecture Notes

Chapter 6:
Wireless

CYENG 351: Embedded Secure Networking

Instructor: Dr. Shayan (Sean) Taheri

Gannon University (GU)

Wireless Technologies

- Wireless applications essentially combine all of the reasons we need security for embedded systems with **limited resources**.
- By definition, many wireless devices will require limited resources, because they will be designed to run on batteries or in environments where available resources must be given to **the communications hardware**.
- In the world of inexpensive embedded systems (we exclude inexpensive consumer products from this category; we are referring more to **industrial-type controllers**), wireless technologies are only just starting to make inroads into the industry.
- As wireless technologies decrease in cost and their implementations increase in number, we will begin to see more and more wireless devices available to smaller companies and hobbyists.
- Wireless is a radical change in the way devices communicate, so this infusion of technology will present some interesting challenges to embedded developers.
- Wi-Fi, which follows **the IEEE 802.11 standard protocols**, has been around for a while, but the technology was typically reserved for consumer applications and PC's - the infrastructure and physical characteristics of the hardware just did not work with many embedded applications.

Wireless Technologies (Cont.)

- We are just now seeing the 802.11 protocols make their way into the lowest reaches of the embedded realm, but Wi-Fi is by no means the only wireless technology available.
- While **the Wi-Fi protocols** dominate the computer industry, cellular communications is probably the most recognizable form of wireless technology.
- **Cellular communications technologies** have been around for a long time, and in some sense, cellular was one of the original "embedded" wireless protocols, since cell phones pretty much embody the principles of an embedded paradigm.
- **Cellular communications** is relatively old in the world of technology, but it was optimized for a single application - **voice-based telecommunications**.
- The cellular networking infrastructure is **stable** and **ubiquitous** and can be put to great use in applications where none of the other wireless protocols would be effective.
- Cellular technology allows for devices to be connected to the global Internet from just about anywhere, but sometimes the scale (and cost) of cellular isn't needed or even justifiable for an embedded application.

Wireless Technologies (Cont.)

- In recent years, **two technologies** have risen to the forefront of wireless communications specifically for the embedded world: **Bluetooth** and **ZigBee**.
- **Bluetooth**, which is the technology behind hands-free mobile phone headsets, PDA keyboards, and a host of other consumer devices.
- **ZigBee** is an 802 protocol, like Ethernet and Wi-Fi (802.15.4 to be exact).
- It was specifically designed for low-power embedded applications.
- In compare Bluetooth, **ZigBee** is more reminiscent of a wireless version of good old RS232 serial port.
- ZigBee is a relatively new protocol (the standard is still in the process of being finalized), but it is already being put to use in some exciting applications.

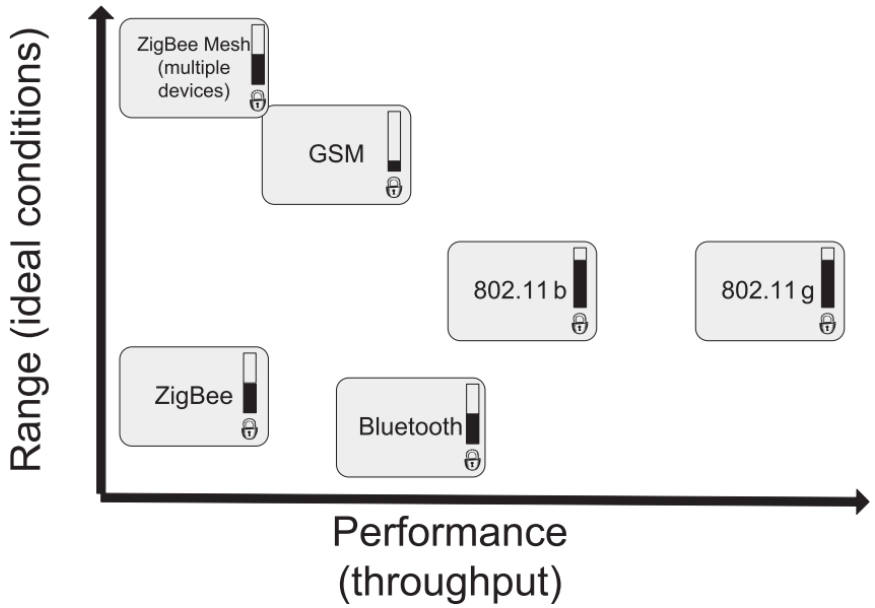
Cellular Technologies

- **Cellular wireless technologies** were created for mobile telephone communications but, like their wired counterparts, have diversified and evolved into **general-purpose** communications technologies.
- A few technologies are of interest when discussing **the connection between cellular networks and digital communications**, including GSM4 (Global System for Mobile communications, the base technology for a majority of cellular communications) and GPRS5(General Packet Radio Service), which adds data transfer capabilities to GSM and allows for services like text messaging and data communications.
- One of the largest barriers to **using cellular technologies** for inexpensive wireless communications (in our case, for embedded control applications) is that cellular networks are difficult to get on to, usually requiring a partnership with the organization that owns the network.
- You can buy GPRS/GSM modems that will allow your application to be connected to a cellular network (the modem vendor will usually have a partnership with at least one or two carriers).
- The closed nature of cellular networks makes security a difficult problem.
- The GSM and GPRS technologies have security built into their specifications, but the methods used are not the best.
- Poor encryption algorithms and questionable security design considerations mean that cellular communications may not be as secure as they could be.

Cellular Technologies (Cont.)

- If you are going to use GPRS/GSM as a communications medium, it is recommended that you use a higher-level security protocol (**SSL** is a good choice) on top of the communications channel.
- Cellular networking allows for a couple of features that are interesting for embedded applications.
- The networks are available nearly everywhere, so a cellular-enabled device would have a network connection nearly anywhere, and cellular networks are very good at providing roaming connections, so devices can move around.
- However, for a large number of embedded control applications, cellular technology is probably overkill.
- If the embedded device is in a warehouse somewhere and does not move around too much, but needs wireless connectivity, since wires are difficult to run, cellular is probably too slow (dial-up modem speeds are normal) or expensive.
- **Our Interest**: More practical wireless technologies for limited-resource applications (and as a bonus, they all happen to be generally easier to secure than GPRS/GSM).

Comparison of Wi-Fi, ZigBee, Bluetooth, and Celluar/GSM



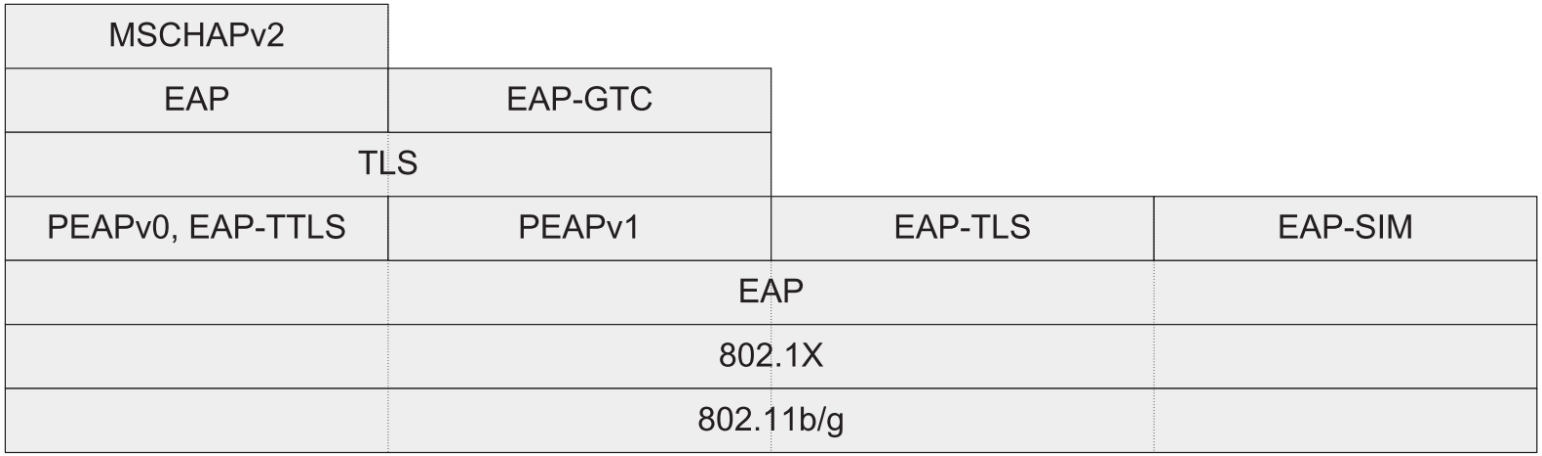
802.11 – Wi Fi

- The **wireless communications technologies** usually referred to by the (trademarked) term "Wi-Fi" are those technologies based off of the **IEEE 802.11 standards**.
- Intended as a general wireless communications protocol (think of Ethernet without wires), 802.11 implementations are by far the most common form of wireless communication between PCs.
- Wi-Fi is characterized by having a medium range of communications capability (as compared to cellular) with a very large (relative) data rate.
- **802.11 wireless** is a heavy-duty wireless protocol, supporting speeds that **rival wired Ethernet** (802.11b is capable of 11Mbits/second, and 802.11g is capable of 54Mbps).
- Its designers recognized the need for security early on and included a security protocol in the original specification: **Wired Equivalent Privacy (WEP)**.
- However, WEP was inherently flawed, due to the use of stream ciphers without accounting for some of the important issues inherent in using stream ciphers.
- One of the major issues was the use of short initialization vectors and infrequent changes of the master RC4 keys.
- **Wi-Fi Protected Access (WPA)** improved upon WEP by addressing the most grievous flaws exhibited by the original protocol, but retained a level of backward-compatibility that allowed WPA to easily be implemented for most systems that previously relied on WEP.
- Authentication was also a problem with WEP, which used the **WEP** key itself, so **WPA** uses a separate authentication protocol.
- Both **WPA and WPA2** improve security by moving to AES (and also supporting various key sizes larger than 128 bits as required by the US government), and differentiating between personal and enterprise networks, which have different requirements.

WPA Key Management

- Authentication in WPA (and WPA2) is utilized to prevent unauthorized connections to a network and to help mitigate threats from **Rogue Access Points - RAP** (so-called "evil twins" that trick the client into believing it has connected to the correct network).
- For **WPA-Personal**, authentication is not strictly required because of the work required to manage an authentication server.
- For WPA-Personal, the **Pre-Shared Key (PSK)** is usually considered enough for authenticating home wireless networks, but the stronger methods could be used if desired.
- In cryptography, a **pre-shared key (PSK)** is a shared secret which was previously shared between the two parties using some secure channel before it needs to be used.
- **For Enterprise authentication**, both **WPA and WPA2** utilize the same basic framework: 802.1X/EAP.
- The **802.1X protocol** (note that the **"X" is really an X**, not a placeholder for a number) is part of the IEEE standards for managing both wired and wireless networks, and defines a secure transport mechanism for **Extensible Authentication Protocol (EAP)** messages.
- EAP provides the basis for authentication in both WPA and WPA2.
- There are a total of 5 variants required to be compliant: EAP-TLS, EAP-TTLS/MSCHAPv28 (TTLS is simply "Tunnelled TLS"), PEAPv0/MSCHAPv2 (Protected EAP, which establishes a TLS connection over which EAP methods are used), PEAPv1/EAP-GTC (an EAP variant developed Cisco), and EAP-SIM which is essentially authentication using SIM cards for the telecom industry.
- In any case, Wi-Fi authentication is a dynamic and complex field and keeping up with it can be quite a challenge (by the time you are reading this it is likely that there have been a number of new protocols added and compliance requirements have changed).

WPA Key Management (Cont.)



Wi-Fi Authentication Mechanisms

Drowning in Acronyms

- We need to adapt the protocol to our application.
- If you are implementing the latest and greatest consumer gadget and you must be compliant with **every wireless access point** (under the sun), you will likely need to implement most or all of the protocols and mechanisms (or more).
- If you are working on a proprietary solution for a specific purpose and you can control what access points are used and just want to have some level of security for your embedded devices, then you can probably scale back to a lower level of authentication.
- In fact, for many embedded applications (especially those with strict budget limits), **WPA-PSK** may be sufficient.
- The **full WPA-Enterprise authentication suite** was designed for large organizations with numerous high-power devices such as laptops and expensive PDA's that need to be continuously updated.
- The level of security provided by an authentication server is probably overkill for an application that monitors the output of an oil well.
- Another important point that has not been addressed is the fact that Wi-Fi is a high throughput, and therefore high-power wireless protocol.
- It is very likely that if you are developing an application using a low-power inexpensive microcontroller, the amount of power required for the 802.11 radio probably exceeds your requirements.

Do You Really Need 802.11?

- The extensive requirements of Wi-Fi, **both in software support and in power consumption**, make Wi-Fi a less attractive option for limited-resource systems.
- It is possible to implement 802.11 wireless for inexpensive systems, but the functionality will likely need to be reduced to meet the system specs.
- Fortunately, more than a few people recognized the need for wireless protocols that provide connectivity without the resource requirements of full Wi-Fi.
- **Two protocols** have risen in recent years that promise the level of connectivity needed by **low-power** and **inexpensive** devices without having to support numerous security protocols and without having the power consumption associated with the higher bandwidth 802.11-based protocols.
- The first protocol is **Bluetooth**, a standard that has come to be a household word due to its widespread use in mobile telephone headsets and various other consumer devices.
- The second protocol is a relative newcomer (the standard is still in the process of being ratified) is **ZigBee**.
- Bluetooth provides a medium level of throughput and is suited for consumer applications.
- ZigBee is tailored specifically for embedded industrial applications which often have radically different requirements than consumer applications.

Bluetooth

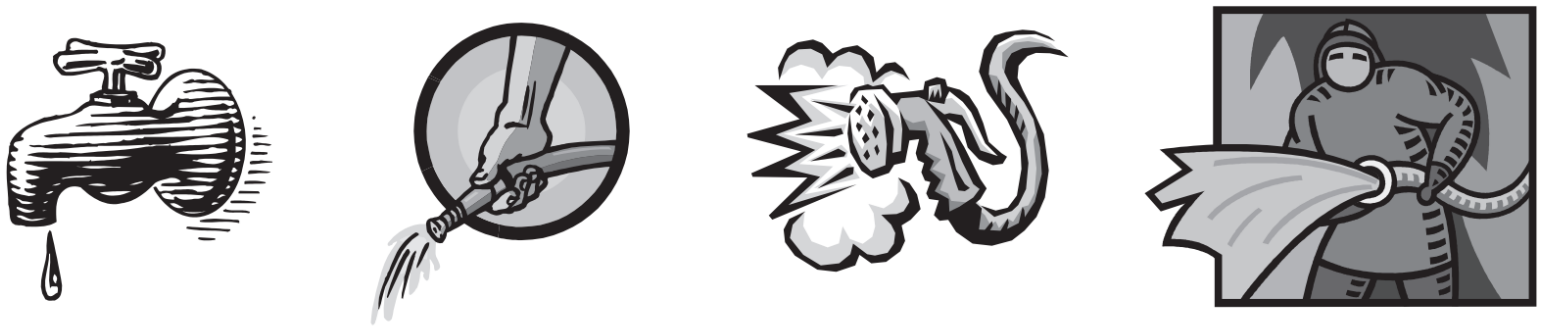
- **Bluetooth** was one of the first wireless protocols to address the power consumption issues that are inherent in **battery-powered consumer devices**.
- **By reducing the bandwidth and range requirements**, the Bluetooth protocol lends itself to battery-powered applications that require a moderate level of throughput, such as wireless headsets for mobile phones and input devices (such as key boards and mice) for PDAs. → Using **the 2.4 GHz spectrum band, 2400 to 2483.5 MHz** → Which enabling a good balance between range and throughput.
- Driven by widespread use, the Bluetooth physical layer specification was adapted by the IEEE to develop the 802.15.1 standard.
- The standard was developed and is controlled by the Bluetooth Special Interest Group - SIG (www.bluetooth.com), and the security is based on **a 3-mode model**, with **an unsecured mode**, **a "service level" secured mode**, and **a link-level secured mode** (the entire connection is secured).
- According to the Bluetooth SIG, all known attacks against the Bluetooth protocol are actually against specific implementations and the protocol itself is secure.
- The security of Bluetooth uses the concept of **two separate keys**, **an authentication key** and **an encryption key**.
- The authentication key is the master key, and encryption keys are regenerated with each new session.
- A random number, generated for each transaction, adds additional security.
- The **bluejacking attack** simply involves the sending of an unwanted message to the device user, which could be used to trick the user into providing sensitive information to the attacker (**phishing**).
- Bluetooth provides a decent midrange protocol for embedded systems that need a moderate level of throughput, but it is a complex protocol and the cost of a dedicated controller unit may be prohibitive depending on the application.
- Bluetooth, being designed for consumer applications, focuses on higher bandwidth and convenience.

ZigBee

- Like the **Wi-Fi Alliance** and the Bluetooth SIG, the ZigBee Alliance is a consortium of corporations that all utilize the protocol.
- **ZigBee** is characterized by **low power consumption** (able to run on batteries for extended periods of time due to the low duty cycle of its radio), **low system resource requirements, and low throughput**.
- The bandwidth of ZigBee is comparable to that of a dial-up modem (**Up to 250KB/s = 0.25MHz**).
- ZigBee is primarily concerned with flexibility of the network (ZigBee supports several network topologies that increase reliability of the entire network) and conservation of resources, especially power consumption.
- **Geared toward industrial automation** (as opposed to consumer connectivity like Bluetooth), **ZigBee is definitely an industrial standard**.
- ZigBee allows for thousands of nodes to be included in a single network, usually referred to as a **Personal Area Network (PAN)**, which was coined to describe the networks specified in the IEEE 802.15 standards (ZigBee radios conform to 802.15.4).
- The interesting thing about some of the topologies supported by ZigBee is that **they are self-healing**, in that the network is **resilient** and can deal with nodes coming in and out of the network, as would be expected in a noisy (RF noise) industrial environment.
- This **self-healing property** makes redundancy very easy to implement, and this can directly translate into a more secure application.

ZigBee (Cont.)

- In a Wi-Fi network, once **a node is dropped**, it must reestablish the connection, including any authentication required. → The attacker could take out a few key nodes and the entire network goes down.
- With a ZigBee network, there can be many more nodes (ZigBee is cheaper to implement) and if any nodes are dropped due to tampering, the network continues to function.
- The basic functionality of ZigBee is setup to use the radio as little as possible to **conserve power**, so the end result is that the nodes are usually "down" anyway.
- Without the need to keep close synchronization (as is the case with Wi-Fi, for example), ZigBee networks can deal with attacks inherently.

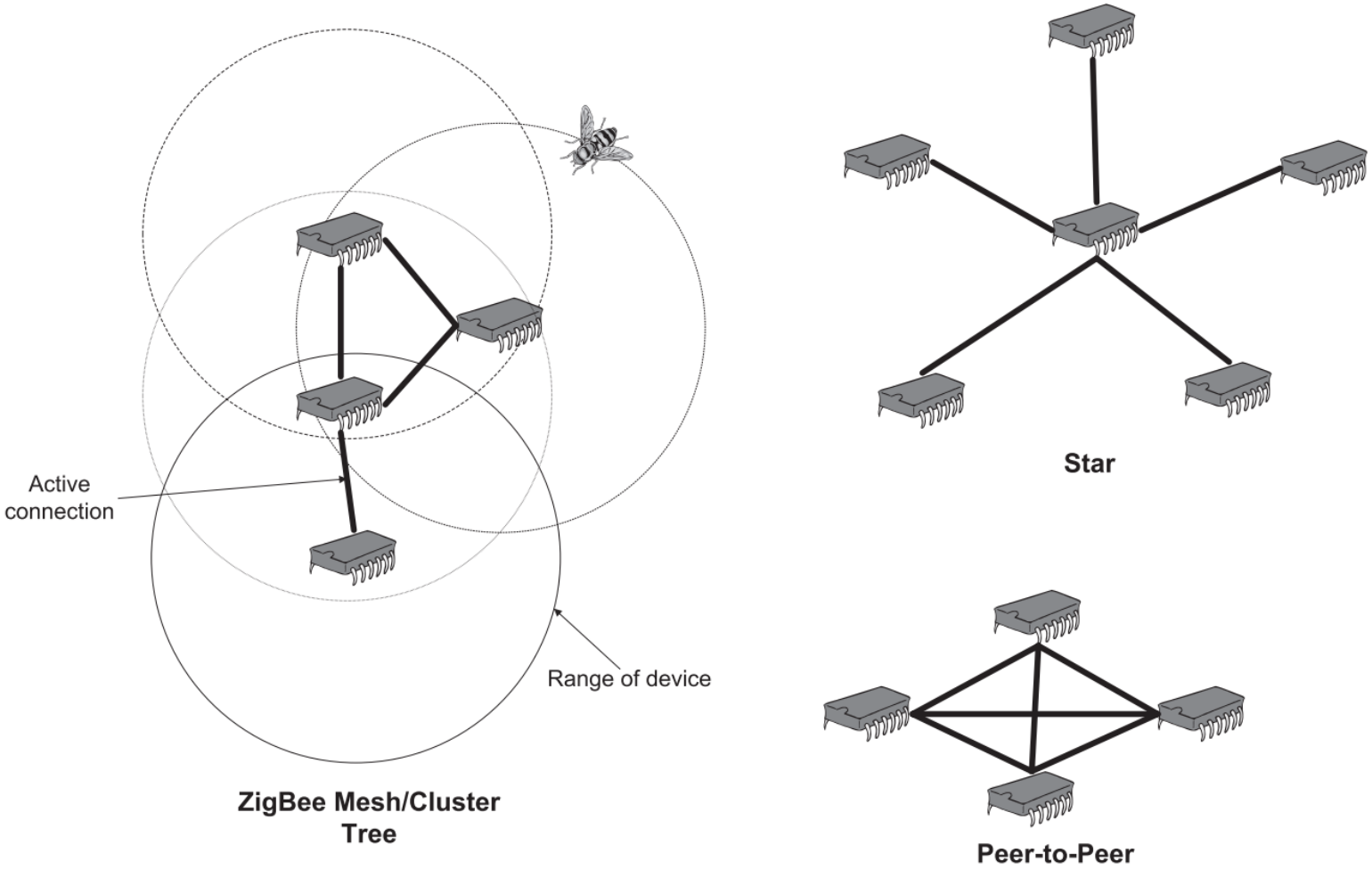


Throughput Comparison

ZigBee (Cont.)

- The **network topologies** supported by ZigBee vary, but depending on the type of network being deployed there are varying security considerations.
- In some ZigBee topologies, all nodes are considered equal and **the network is basically an ad hoc peer-to-peer configuration**.
- In some of the other topologies (**star and tree forms**), there must be a coordinator that facilitates the network. → In such a topology, the end-nodes can be of reduced functionality to save cost, but there is an inherent problem. → If the coordinator is disabled, reduced-functionality end nodes cannot reestablish the network. → One way to get around this is to **provide redundant full-function nodes** that can all serve as a coordinator (this is **a good idea for reliability, not just security**).
- ZigBee nodes can also function as routers, directing communications between **nodes** that may not be able to communicate directly (as would be the case if the distance between those nodes was too great, but the router node bisected the path between them).
- The ZigBee protocol provides **low-level security** for communications between individual nodes using **AES** and **a message authentication code scheme**, but this only protects the data between nodes, not on the nodes themselves.

ZigBee (Cont.)



ZigBee Topologies

Wireless Technologies and the Future

- It is likely that engineers and corporations will push wireless technology to its boundaries, and **security** will be of the utmost importance.
- You should consider **security** when choosing a technology for your particular application.
- When it comes to security, it never hurts to err on the side of caution and use far more computing power than you need to be sure that you can support the security you need both at deployment and years down the road.

Assignment

- **Reading Assignment:**
  - Stapko, T., 2011. **Practical embedded security: building secure resource-constrained systems**. Elsevier.
    - ✓"Chapter 6: Wireless", Pages 115-127.

Questions?