Developing Wi-Fi® Connected IoT Devices

Topics

- Why to Use Wi-Fi for IoT?
- Basics of Wi-Fi
- Developing a Wi-Fi Connected IoT Device

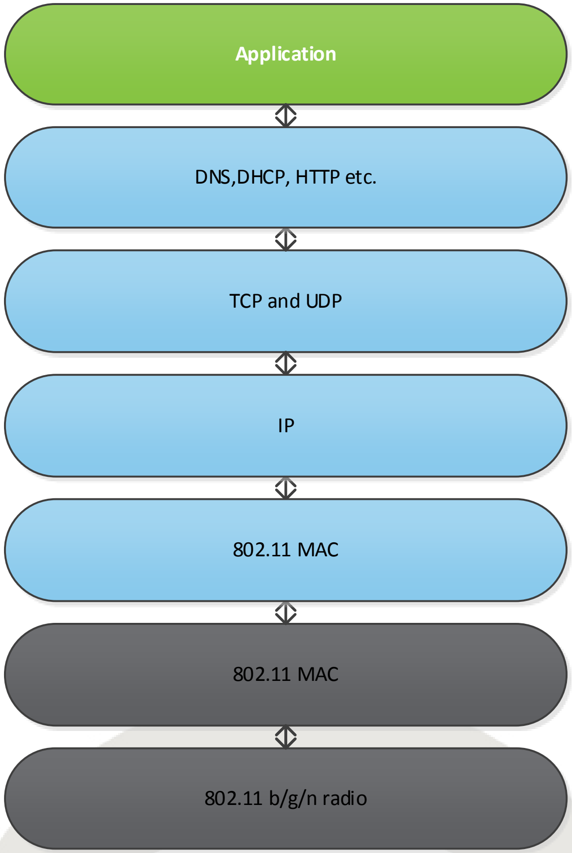
Why to Use Wi-Fi for IoT?

• **Standardized**	- 802.11 and Wi-Fi Alliance
• **Unlicenced frequencies**	- 2.4GHz and 5GHz
• **High speed data**	- MBs to GBs
• **Security**	- WPA2, WPS, WPA Enterprise
• **Relatively low power**	- 5-200mA
• **IP Connectivity**	- IP, TCP and UDP
• **Application level protocols**	- HTTP, DHCP, DNS etc.
• **Installed infrastucture**	- 25 to 80% of homes have Wi-Fi - USA and Europe – 60 to 80%

Basics of Wi-Fi

- **802.11 MAC**
  - Active and passive scanning
  - Authentication and association
  - Encryption
  - Flow control and fragmentation
  - Power saving

- **802.11 Radio**
  - 2.4 and 5GHz
  - DSSS and OFDM modulations
  - 22MHz to 160MHz channel bandwidth
  - 1 – 14 channels
  - 1 – 433Mbps symbol rates

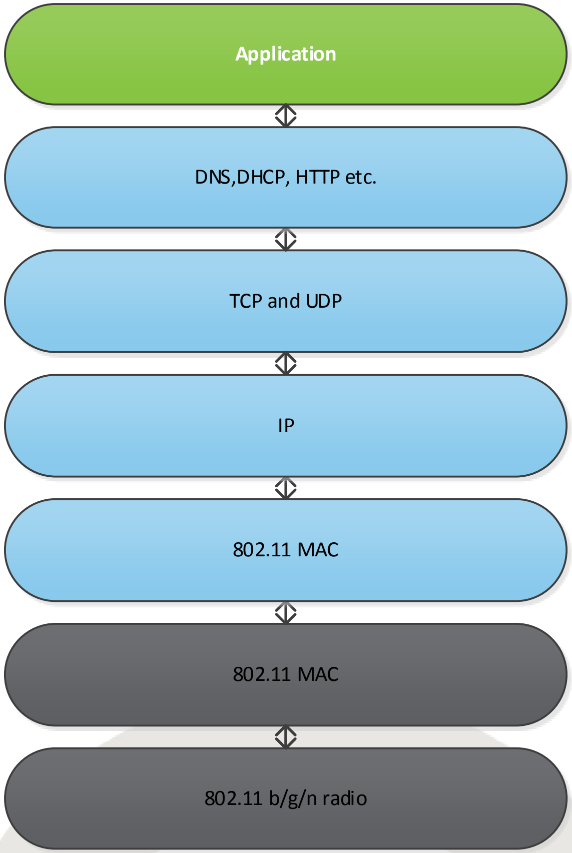


Basics of Wi-Fi

- **Security**
  - Authentication
  - Association
  - Access Control
  - Encryption

- **Encryption options**
  - WPA2 Personal
  - WPA Enterprise
  - WPA
  - WEP
  - Open

- **Wireless Protected Setup (WPS)**
  - Easy security setup with PIN entry or push button



Implementing a Wi-Fi IoT Sensor

Typical Questions to Ask

- What to Connect?
- How to Get Connected?
- Security?
- How to Discover Devices and Services?
- How to Transmit Data?

What to Connect?

- **Internet for exmple via Wi-Fi Access Point**
  - Your device needs to be a Wi-Fi client
  - Wi-Fi Access Point settings (SSID and security) need to be configured in the client

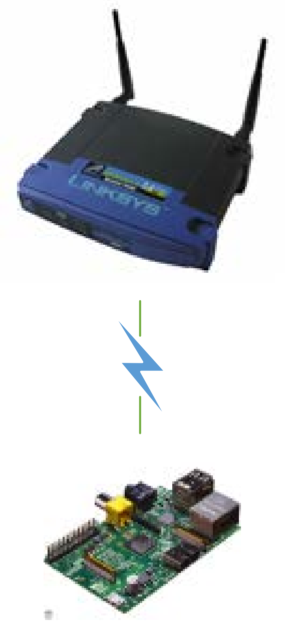
- **Point-to-Point eg. Smart Phone or Tablet**
  - Your device should be a Wi-Fi Access Point
  - You can easily scan and connect it with a smart phone
  - However when you do this the smart phone cannot be connected to connect Internet at the same time

- **Point-to-Point while Smart Phone connected to Internet**
  - Wi-Fi Direct (WFD) allows P2P connection while smart phone connected to Internet
  - WFD however not widely supported on smart phones

How to Get Connected?

Getting to Internet via Wi-Fi Access Point

- **Challenge**: Access Point settings need to be cofigured to the device
- **Configuration options:**
  - WPS and simple Led + button interface
  - AP mode + HTTP server
  - Ethernet + HTTP server
- **Normal operation**
  - DHCP
  - TCP, UDP etc. For data trasfer



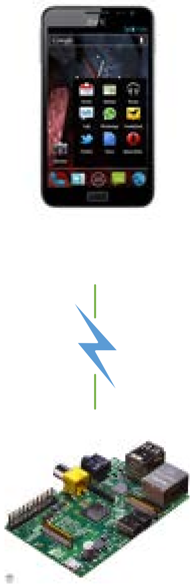
Access Point:
- SSID
- Password
- WPS
- DHCP

Configuration mode:
- Wi-FI AP mode + HTTP server
- Ethernet + HTTP server
- WPS + button press

Operational mode
- DHCP client
- TCP/UDP
- HTTP etc.

How to Get Connected?

- Point-to-Point Connectivity to Smart Phones
  - Relatively simple unless Smart Phone needs Internet connectivity
  - In this case Wi-Fi Direct needed – which is not generally supported yet



Smart Phone

- Wi-Fi client mode
- Use the built-in UI to disocver and connect the device

Internet connection

- Wi-Fi Direct needed or otherwise phone will drop from Internet

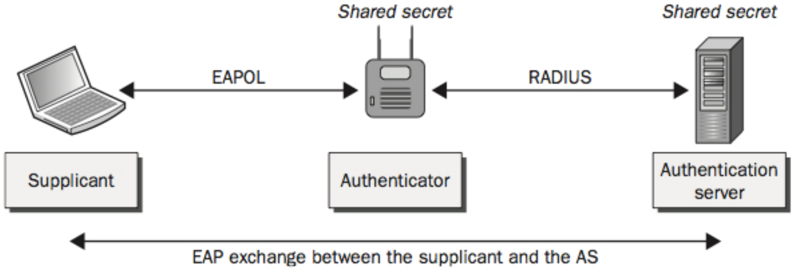
Configuration / Operational mode:

- Wi-Fi Access Point
- HTTP server
- TCP/UDP servers etc.

Security?

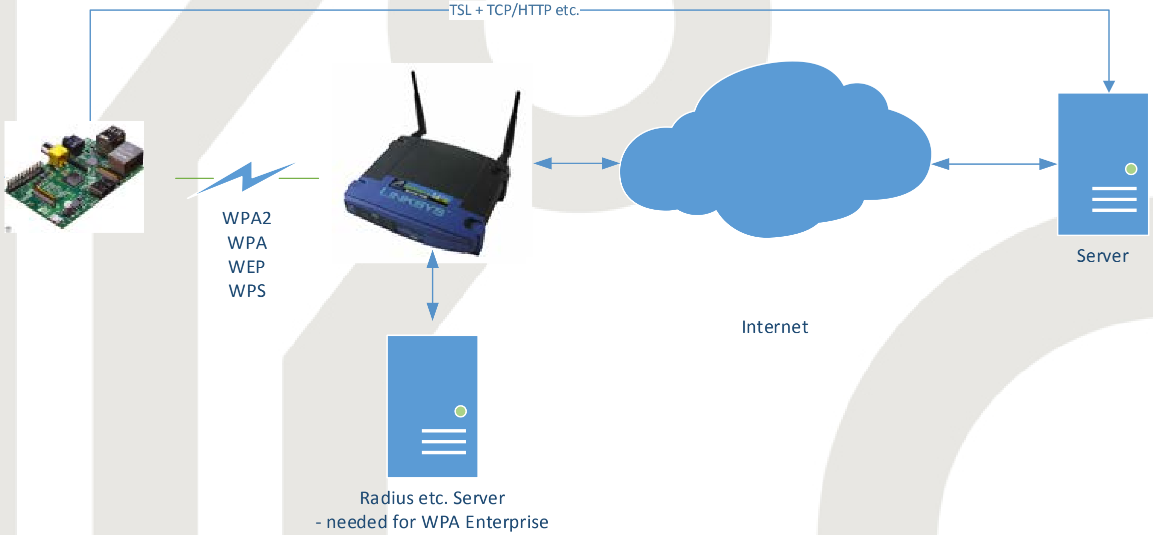
- **Wi-Fi Security**
  - WPA2 is the only secure protocol today and WPA and WEP should not be used at all
  - WPA personal requires a pre shared password to be configured in both the Access Point and the Client
  - Wi-Fi security only provides authentication and encryption between the client and the Access Point

- **Enterprise security**
  - Some enterprise networks use WPA Enterprise and do not simply rely on WPA personal
  - The clients are authenticated to a separate authentication server (f.ex. RADIUS), not just the Access Point
  - Uses EAP protocol (802.11x)
  - **PEAP-MSCHAPv2**
    - Username and password exchanged in a TLS tunnel
  - **EAP-TLS**
    - X.509 certificates used instead of username / password



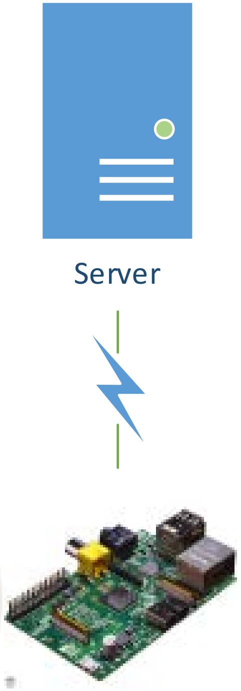
Security?

- **End-to-End security**
- **Transport Layer Security adds end-to-end security over TCP**
  - SSL ius also supported, it is now considered insecure
    - POODLE Attack
    - https://www.us-cert.gov/ncas/alerts/TA14-290A
  - TLS offers two services
    - Verification of the servers identity
    - Encryption of data
  - X.509 certificates are needed at the client and server



How to Discover Devices and Services?

- **Server Discovery**
  - Servers typically have fixed IP address / DNS name
  - Need to be programmed in the application code
  - DNS client can be used to tranlate URLs into IP addresses



Server:

- IP address
- Domain name

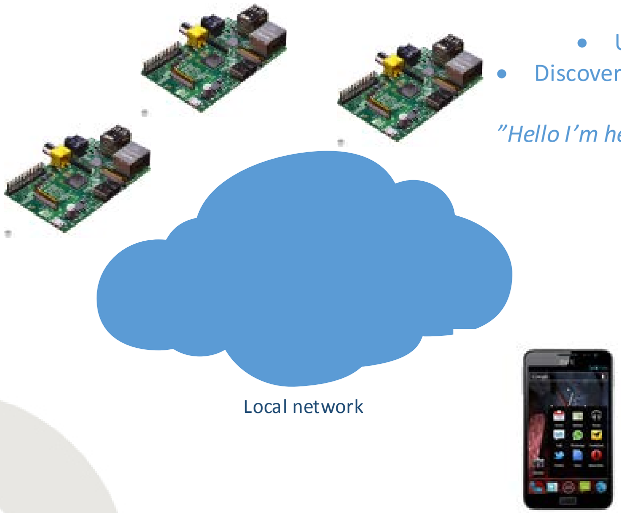
  server.mydomain.com

Client

- Server domain name programmed
- Use DNS to resolve IP

How to Discover Devices and Services?

- **Client discovery**
  - More complex as clients do not neccesarily have fixed IP or DNS name
  - Multiple clients can be in the same network
  - **Discovery strategies:**
    - UDP broadcast / multicast
    - Discovery protocols like mDNS or UPnP
  - mDNS applications are available for iOS and Android devices
  - Note: No-built in support for mDNS or UPnP, but they are fairly trivial to implement over UDP (even with BGScript)



- UDP broadcast
- Discovery protocols like mDNS

"Hello I'm here. I'm an IoT Device"

Local network

- UDP listener (f.ex. mDNS)

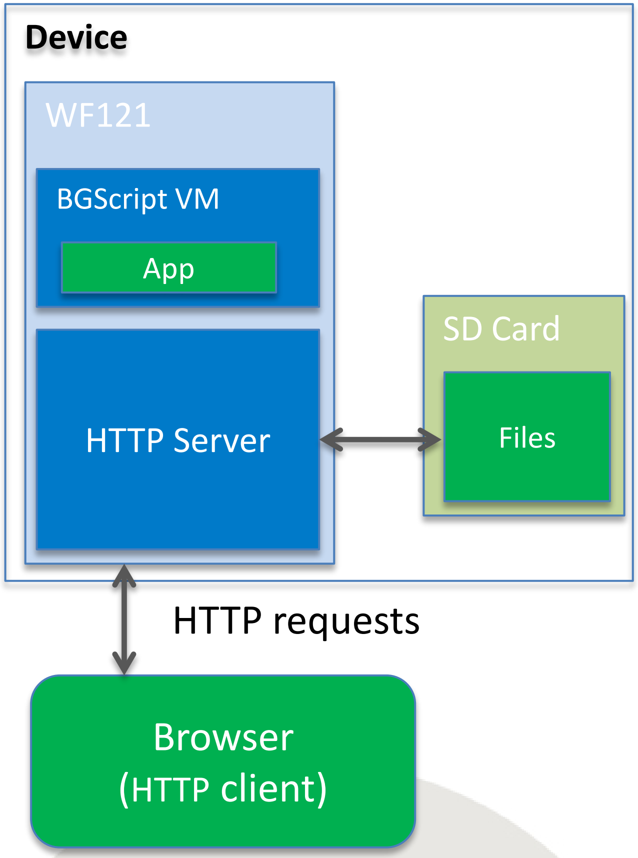
"I can see three IoT devices with
10.x.x.1,
10.x.x.2 and
10.x.x.3"

How to Transmit Data?

- **UDP**
  - Connectionless data transfer
  - Enables broadcast
  - However can be unreliable
  - WF121’s throughput ~3.5Mbps
- **TCP**
  - Connection oriented data trasfer
  - Provides reliability and retransmissions
  - WF121’s throughput ~3.5Mbps
  - Up to 32 TCP sockets
  - Can be secured with TLS
- **HTTP**
  - Browser can be used as an application
  - Allows simple user interfaces to be built with HTML + Javascript

Example: Standalone Temperature Sensor using HTTP

- **Features**:
  - Wi-Fi Access Point Mode
  - WPA2 security
  - DHCP and HTTP servers
  - BGScript application
  - I2C
- HTML files are stored on the WF121s built-in flash
- Alternatively they can be stored on external SD card connected to one of the SPI interfaces
- A temperature sensor connected to the WF121’s I2C interface
- **BGScript Application:**
  - Configures Wi-Fi AP settings
  - Starts AP mode
  - Start DHCP and HTTP servers
- **Reading and displaying the temperature:**
  - Web browser requests URL : /I2C/readtemperature.html
  - An event is generated to BGScript application
  - BGScript application reads temperature over I2C
  - BGScript application returns the respons as HTML page or JSON file



Thank You