WIRELESS SECURITY

Overview

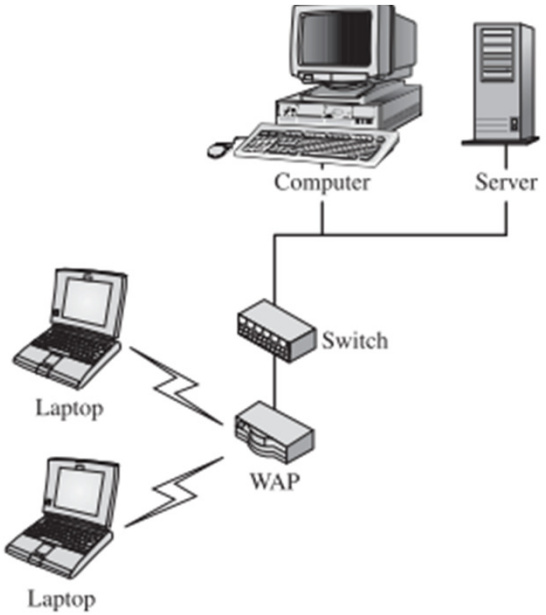
- Wireless technology poses a greater security risk to the data that is transferred because the information is broadcast to anyone within range of the signal

Wireless Networking Standards

- 802.11a : 54Mbps but at a shorter range
- 802.11b: 802.11b supports speeds up to 11Mbps over a longer range than 802.11a.
- 802.11g: 802.11g supports speeds up to 54Mbps and is downward compatible with 802.11b
- 802.11n : 100+ Mbps, improvements over 802.11g
- 802.11ac: 450Mbps→1300Mbps
- ….

Wireless Modes

- Ad hoc mode: a peer-to-peer communication mode, in that clients communicate directly with each other
- Infrastructure mode: the clients connect to a wireless device called a wireless access point



Preventing Invalid Connections

- Service Set Identifier (SSID):
  - Only clients that have been configured with the same SSID as each other or the access point can connect
  - But, most access points will broadcast the SSID to all the clients for ease of configuration, can turn off the broadcast so that the access point runs in stealth mode
- MAC Address Filtering:
  - Require to set up MAC filtering to specify which clients want to allow to connect
  - Attackers can use MAC address spoofing to overcome this restriction

Wired Equivalent Privacy

- WEP: Provide authentication and encryption on any infrastructure-mode wireless network
- WEP Encryption: using RC4 symmetric key encryption to provide for the encryption of transmitted data
- WEP Authentication: provides for open and shared key authentication
- AirSnort can be used to break WEP encryption and read transmitted messages

Wi-Fi Protected Access (WPA)

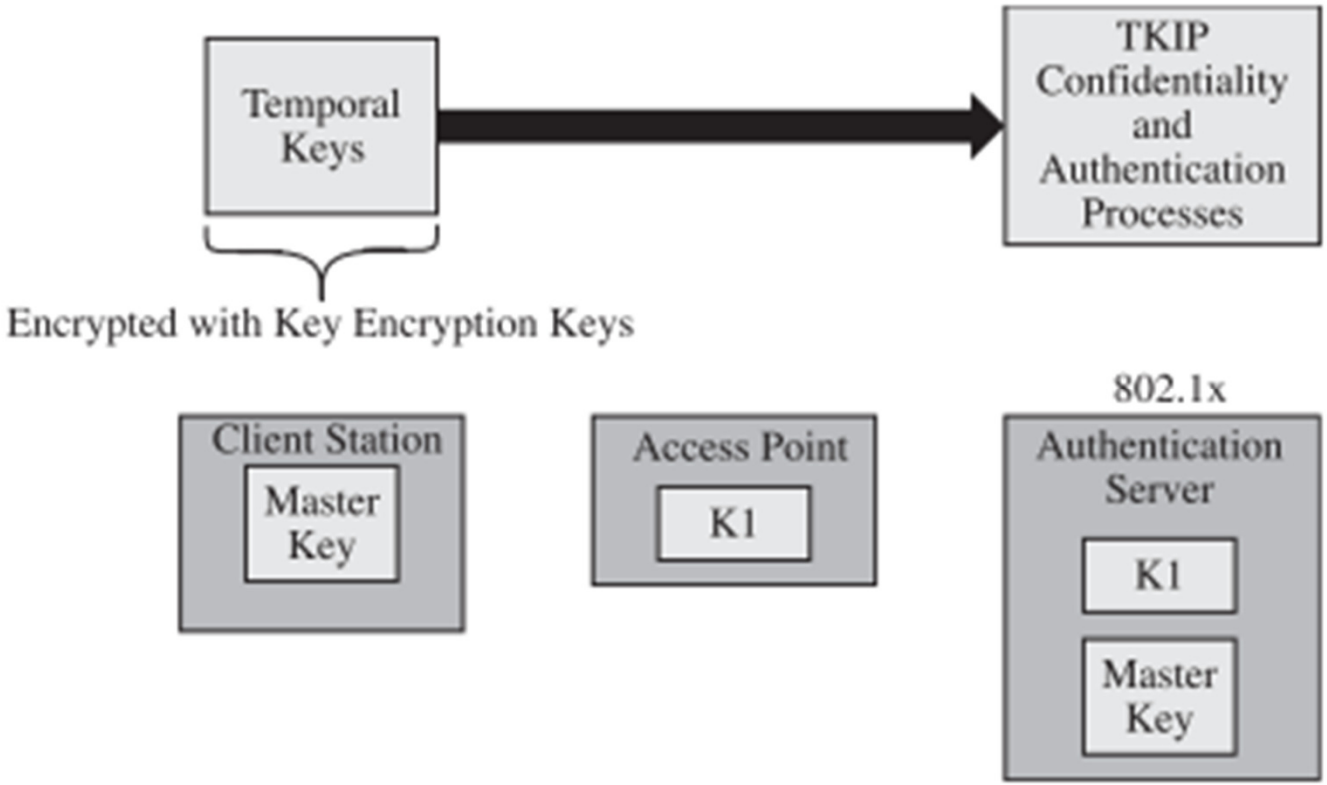
- Was developed to eliminate some of the vulnerabilities of WEP.
- Use of EAP for authentication and Temporal Key Integrity Protocol (TKIP) to provide message integrity
- TKIP is a strategy for managing encryption keys that provides some improvements to help protect against the vulnerabilities in WEP. These enhancements:
  - Per-packet Mixing Function
  - IV Sequencing Discipline
  - Message Integrity Codes (MICs)

Standard 802.1x

- 802.1x standard was developed to help administrators provide greater security to wireless networks
- Using TKIP to manage data integrity, and
- Using EAP-TLS to authenticate the user
- EAP-TLS requires a PKI to manage the creation, distribution, and revocation of certificates.
- The 802.1x standard also provides for encryption of each connection using TLS. This means that the keys for encryption can be negotiated per session.

Standard 802.1x : Re-keying Against Key Reuse

- To protect against key reuse, 802.1x uses a hierarchy of master keys, key encryption keys, and temporal keys
- The 802.1x temporal keys are used in the TKIP authentication and confidentiality processes.
- The material used to generate the temporal keys must be protected from compromise; this protection is accomplished using key encryption keys
- The master key is needed to set up the key encryption keys.



The relationships and locations of the three types of keys

▲ 802.1x provides that the authentication server and client station share a secret key, the master key.

▲ 802.1x further provides that the authentication server and access point share a secret key, derived by the authentication server and client station from the master key and distributed by the authentication server to the access point.

▲ A new master key is used with each session (a session covers the time from when the user is authenticated to when the key expires, when the key is revoked, or when a client station no longer communicates).

▲ The master key is used to protect the communication of key encryption keys between a client station and the access point.

▲ The key encryption keys are employed to protect the transmitted keying material used by the access point and client to generate sets of temporal keys.

▲ The pairs of temporal keys are used for integrity protection and confidentiality of the data.

Configuring 802.1x in Active Directory

- In an Active Directory environment, you can configure 802.1x through Group Policy Security Settings
- Allows you to manage many clients at once with Active Directory, which will solve some of the management problems with 802.11

Configuring steps in Windows

1. Open the Security Settings section of Group Policy by navigating to the Domain Security Policy Microsoft Management Console (MMC).

2. Can use the Wireless Network (IEEE 802.11) Policies node to configure 802.11 and 802.1x configuration settings Right-click the Wireless Network (IEEE 802.11) Policies node and choose Create Wireless Network Policy from the context menu to configure a wireless policy.

3.This will launch a Wireless Network Policy Wizard that will ask you to enter the name of the wireless policy and then ask if you would like to edit the wireless policy. Give the policy a meaningful name and then click Next.

4. Click the Finish button to end the Wireless Network Policy Wizard and reveal the Wireless Network Policy Properties dialog box.

Using Protected Extensible Authentication Protocol

- PEAP can be used with usernames and passwords, one-time passwords, and token cards. PEAP is supported by Active Directory, Novell®NetWare® Directory Service (NDS), Lightweight Directory Access Protocol (LDAP) directory services, and one-time password database systems.
- PEAP allows the client to use a password to authenticate the user on the wireless network

Standard 802.11i

- The 802.11i wireless security standard incorporates TKIP, 802.1x, and the Advanced Encryption Standard (AES)
- It uses the following set of keys:

▲ A symmetric master key: This key is known by the authentication server and client station for the positive access decision.

▲ A pairwise master key (PMK): This key is a fresh symmetric key known by the access point and client station and is used for authorization to access the 802.11 medium.

▲ A pairwise transient key (PTK): This key is a collection of the following operational keys:

- ▲ Key encryption key (KEK): This key is used to distribute the group transient key (GTK), which is an operational temporal key used to protect multicast and broadcast data.
- ▲ Key confirmation key (KCK): This key binds the PMK to the client station and access point.
- ▲ Temporal key (TK): This key protects transmitted data and varies with time.

WPA-2

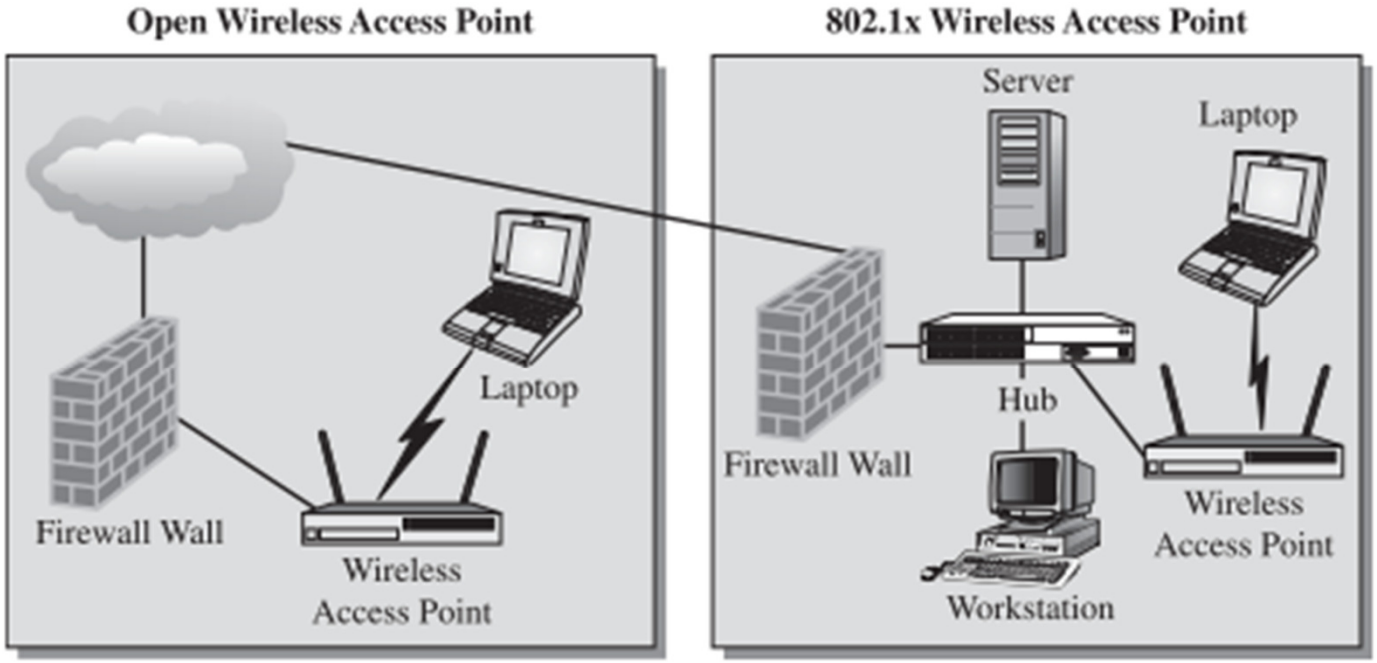
- WPA2 is used on all certified Wi-Fi hardware since 2006 and is based on the IEEE 802.11i technology standard for data encryption
- The major difference between WPA2 and WPA is that WPA2 further improves the security of a network because it requires using a stronger encryption method called AES

KRACK Attack

- **K**ey **R**einstallation **A**tta**ck**s, breaking WPA2
- *Discovered by Mathy Vanhoef,* Computer and Communications Security (CCS) conference, and at the Black Hat Europe conference, on 1st November 2017
- Main attack is against the 4-way handshake of the WPA2 protocol
- This handshake is executed when a client wants to join a protected Wi-Fi network, and is used to confirm that both the client and access point possess the correct credentials (e.g. the pre-shared password of the network).
- The 4-way handshake also negotiates a fresh encryption key that will be used to encrypt all subsequent traffic

Designing for an Open Access Point

- If want to provide open access to the Internet for clients, consultants, or the general public. To do so, should use a network that is shielded from internal network by a firewall.
- Should also control the types of traffic that could be passed to the Internet on the open wireless access point by using a router, firewall to prevent abuse of the public system



Open access point.

Identifying Wireless Network Vulnerabilities

- Vulnerabilities need to consider:

▲ WEP keys must be manually configured in many devices and there is no standard to manage them. You typically have to set them up on the client manually.

▲ Packet checksums, which are the result of a mathematical calculation on the packet that is added to it to verify the integrity of the packet, are not encrypted, so an attacker can manipulate the packets in transit.

▲ The destination or source of a packet can be changed.

▲ Shared key authentication is all that is available without 802.1x.

▲ There is no user or machine authentication option with 802.11 protocols, so you only need to know the SSID to connect (if WEP is not enabled).

▲ Many access points have a well-known default SSID. For example, a LinkSys®wireless access point's default SSID is LINKSYS. An attacker will guess the defaults on popular devices first to determine if they can gain access.

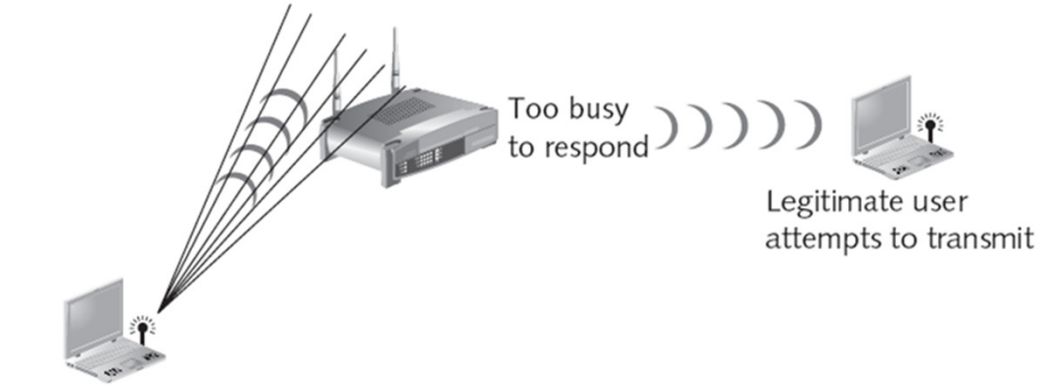
Main threats to a wireless infrastructure

▲ Attackers can eavesdrop on wireless packets because they are broadcast through the air.

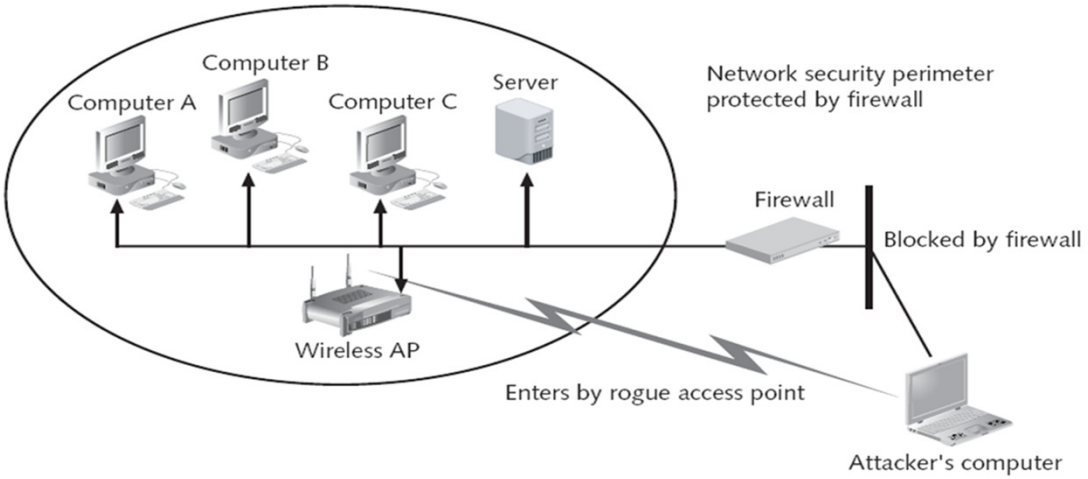
▲ Employees or attackers can add unauthorized access points to a network to provide access to it. These access points normally will not be secure, opening up a weakness on your network.

▲ Denial-of-service (DoS) attacks can be launched by broadcasting a stronger signal, jamming the air with noise, redirecting packets, or disconnecting clients.

▲ Attackers can figure out your SSID or valid MAC addresses by intercepting wireless packets even if you disable SSID broadcasting or enable MAC filtering.



Wireless DoS attack



Rogue access point

802.1x and RADIUS (1/2)

- One of the big downsides of using WPA: require all valid users to know the key to the network
- 802.1X, a port-based authentication protocol originally developed for use on Ethernet LANs to control access to physical ports on a switch
- 802.1X came into play with wireless LANs when work began on the 802.11i standard
- 802.1X makes use of a Remote Access Dial-In User Service (RADIUS) server to provide authentication, authorization, and accounting

802.1x and RADIUS (2/2)

- Other components in an 802.1X-controlled network include the authenticator and the supplicant.
  - The authenticator is the device that provides access to the network's resources (e.g., a switch or AP). When a device is connected to the network, the authenticator detects it and asks it to identify itself.
  - The supplicant is a piece of software on the connecting device that responds.
- The authenticator then acts as an intermediary between the supplicant and the authentication server until access is granted

Deploying the RADIUS Server

- FreeRADIUS (http://www.freeradius.org)
- Downloading; unpack and change into the directory that it creates; build it by running ./configure && make. After it finishes, become root and run make install.
- Need to create a user and group for it to run under, _radiusd
- Edit FreeRADIUS's configuration file, radiusd.conf, in /usr/local/etc/raddb
- add a couple of lines similar to these:

  user = _radiusd

  group = _radiusd

Deploying the RADIUS Server

- Edit the eap.conf file in the same directory and locate the following line in the eap section:

  default_eap_type = md5

- change it to :

  default_eap_type = peap

- If don't already have a Certificate Authority, create one and generate a certificate/key pair for the authentication server, also distribute the CA's certificate to your clients

Deploying the RADIUS Server

- uncomment the tls section and set all of the certificate variables to point to your server's certificate, key, and CA certificate files
- Also uncomment the following lines:

  dh_file = ${raddbdir}/certs/dh

  random_file = ${raddbdir}/certs/random
- Uncomment the peap section and then uncomment the following line:

  # default_eap_type = mschapv2

Deploying the RADIUS Server

- To allow the authenticator to access it, by editing clients.conf and adding an entry similar to this:

```
client 192.168.0.5 {
secret = authpass
shortname = openwrt-ap
}
```

secret is a password that the authenticator will use to access the server and **shortname** is a short descriptive name for the device

- To add users to the RADIUS server, edit the users file and add entries like this:

  andrew User-Password == "wlanpass”

- Need to change the owner of radiusd’s *log* and *run* directories to the user that you created:

  # chown _radiusd /usr/local/var/log/radius

  # chown _radiusd /usr/local/var/run/radiusd

- Then, can start radiusd:

  # /usr/local/sbin/radiusd

Configuring AP for RADIUS

- If AP supports 802.1X, there should be a WPA Enterprise, WPA2 Enterprise, or 802.1X setting in the section of the device's configuration interface.
- Once change it to use 802.1X, you'll need to tell your AP the IP address of your RADIUS server and the password to use when talking to it

Deploy a Captive Portal

- Introduction
- Users try to access the Internet through wifi network are redirected to a web page where they can register for an account that is linked to an email address
- WiFiDog (http://wifidog.org): a flexible portal, consists of a central authentication server and a gateway component that can be deployed on an Access Point running OpenWRT (http://openwrt.org)

WiFiDog: The Authentication Server (1/6)

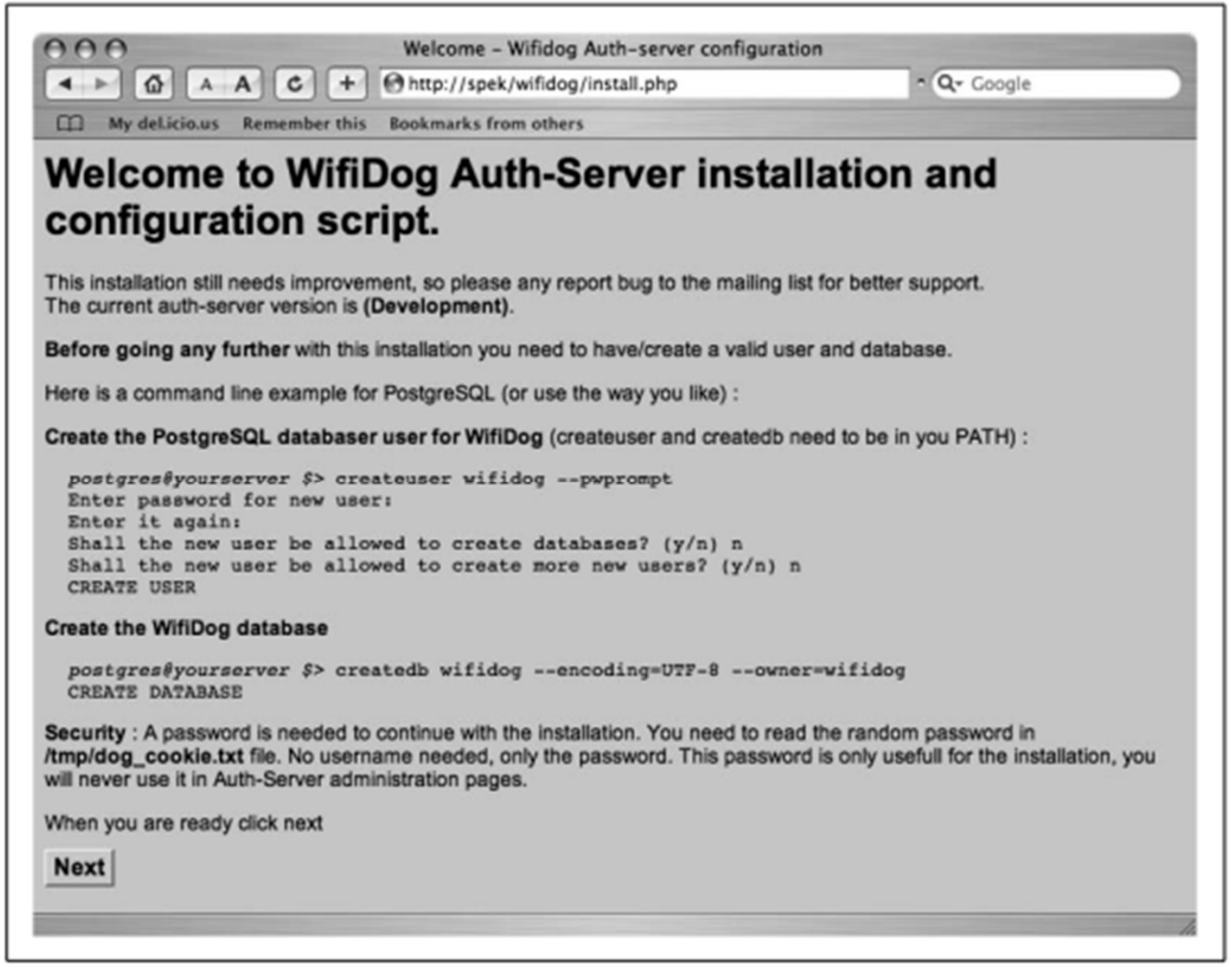
- Set up a PostgreSQL (http://www.postgresql.org) database server. This doesn't need to be on the same machine as web server, but it can be.
- Need PHP 5.x (http://www.php.net) installed on your web server as well
- Checking out the source code from the project's Subversion repository:

```
$ svn checkout https://dev.wifidog.org/svn/trunk/wifidog-auth
```

WiFiDog: The Authentication Server (2/6)

- Change into the directory that it created and move the contents of the wifidog directory along with the sql directory to an area on your web server capable of executing PHP scripts
- Browse to the URL corresponding to where you put the files

WiFiDog: The Authentication Server (3/6)

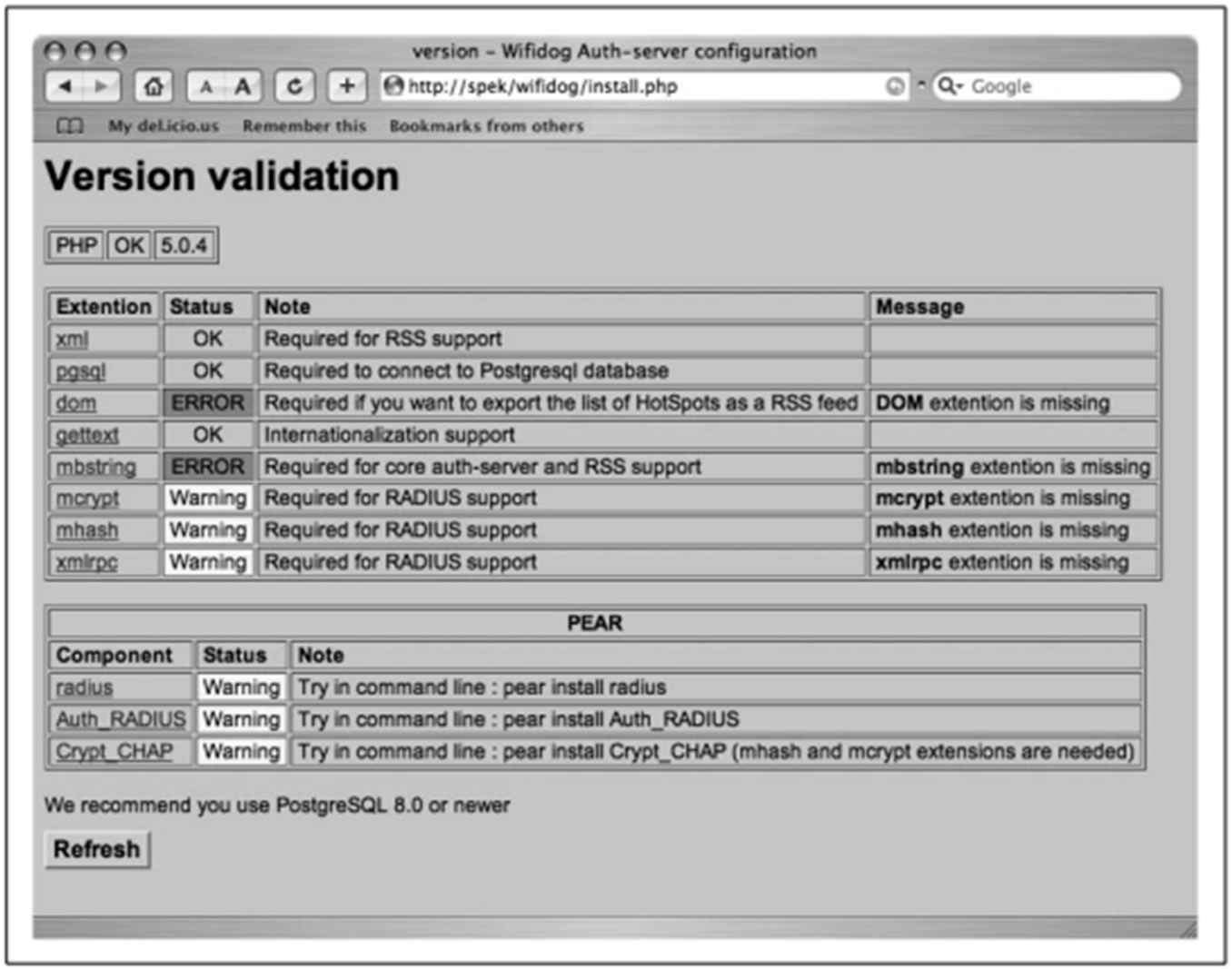


The authentication server installation wizard

WiFiDog: The Authentication Server (4/6)

- Follow the instructions on the page to create the PostgreSQL database. After created the database and a user to access it, click Next.
- In order to proceed, will need to enter the password found in /tmp/dog_cookie.txt on the web server
- Should see something similar to :

WiFiDog: The Authentication Server (5/6)



Status of prerequisite software packages

WiFiDog: The Authentication Server (6/6)

The authentication server's home page

WiFiDog: Installing the Gateway (1/4)

- Log into your OpenWRT-based AP
- Download the WiFiDog gateway package from http://www.ilesansfil.org/dist/wifidog/bin/openwrt/, and run the following commands:

```
# cd /tmp
# wget http://www.ilesansfil.org/dist/wifidog/bin/openwrt/whiterussian-rc3/
wifidog_1.1.3_beta2-1_mipsel.ipk

# ipkg install wifidog_1.1.3_beta2-1_mipsel.ipk
```

- Make sure also have the libgcc package installed:

```
# ipkg list_installed | grep gcc
libgcc - 3.4.4-8 - GCC support library
```

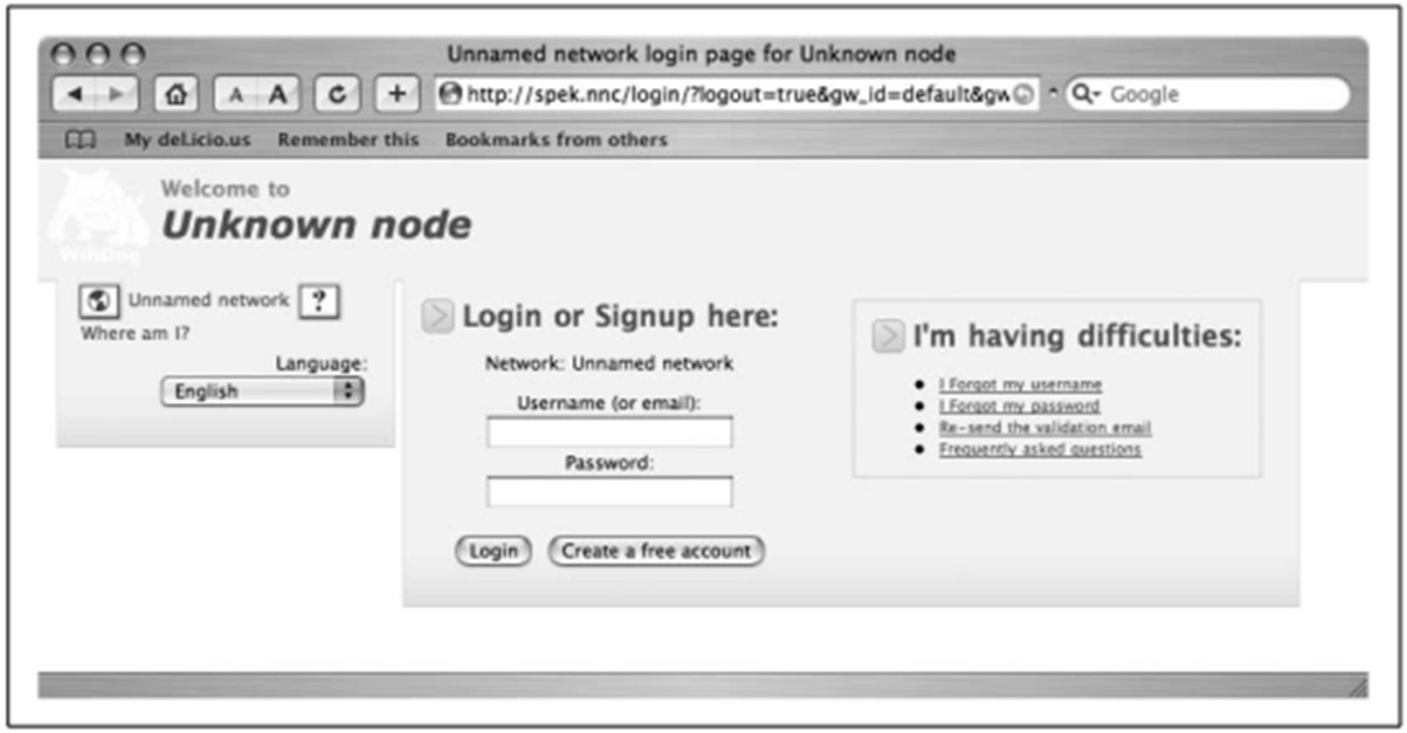
WiFiDog: Installing the Gateway (2/4)

- Edit /etc/wifidog.conf, following the instructions in the file
- Need to tell it where to find the authentication server have been set up, can do this with an AuthServer statement, like so:

```
AuthServer {
  Hostname spek.nnc
  Path /
}
```

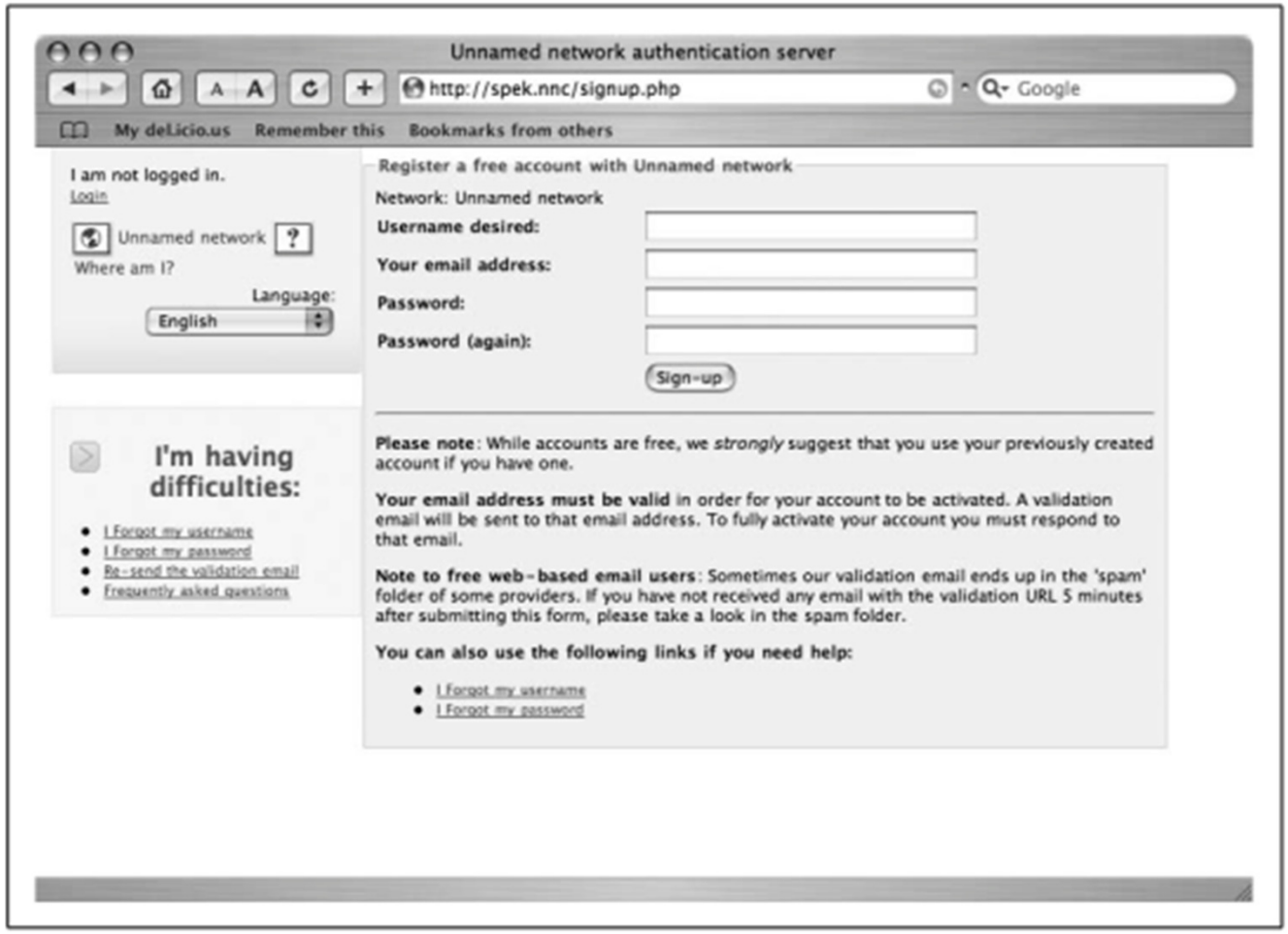
WiFiDog: Installing the Gateway (3/4)

- Reboot the AP
- Try to browse, it automatically redirected to a page that looks like



The login page

WiFiDog: Installing the Gateway (4/4)



Creating a user account

Wireless Hacking Tools (1/2)

- **Aircrack:** It is used as 802.11 WEP and WPA-PSK keys cracking tool, powerful and used most widely across the world. It captures packets of the network and then try to recover password of the network by analyzing packets
- **AirSnort :** wireless LAN password cracking tool. It can crack WEP keys of Wi-Fi802.11b network
- **Kismet :** wireless network sniffer and intrusion detection system, This tool passively collects packets to identify standard network and also detects the hidden network
- **Cain & Able:** It is one of the most popular password cracking tools.
- **Fern WiFi Wireless Cracker:** It helps to see real-time network traffic and identify hosts, to find flaws in computer networks and fixes the detected flaws
- **CoWPAtty:** an automated dictionary attack tool for WPA-PSK to crack the passwords
- **Airjack:**a packet injection tool. It is used to perform DOS attack and MIM attack
- **inSSIDer:** can do various tasks, including finding open Wi-Fi access points, tracking signal strength, and saving logs with GPS records

Wireless Hacking Tools (2/2)

- **Wifiphisher:** This tool can execute fast automated phishing attack against a Wi-Fi wireless network to steal passwords
- **KisMac:** similar to kismet and run on Mac
- **Reaver:** an open-source tool for performing brute force attack against WPS to recover WPA/WPA2 pass keys, https://code.google.com/p/reaver-wps/downloads/list
- **Wifite:** supports cracking WPS encrypted networks via reaver
- **OmniPeek:** packet sniffer and network packets analyzer tool
- **CloudCracker:** CloudCracker is an online password cracking tool to crack WPA keys of Wireless network. This tool can also be used to crack various other kind of password hashes
- **CommonView for Wi-Fi:** a popular wireless network monitor and packer analyzer tool
- **Pyrit :** performs brute-force attack to crack the WPA/WPA-2 passwords, https://code.google.com/p/pyrit/

The End