

Ten things to consider when securing an embedded 802.11 Wi-Fi device

By Timothy Stapko

THE RISKS to users of wireless technology have increased as the popularity of wireless grows. Hackers are becoming more sophisticated so it's very important that wireless device manufacturers secure their devices properly. The top 10 considerations when securing an embedded 802.11 device include:

1) Consider the environment in which the application is deployed.

This is easy to forget, but is vital to the security of your application. Embedded systems are deployed in the real world. Physical security is a big issue for these systems since an attacker can usually compromise a device much more easily with physical access. In addition, the noise created by other applications in the same area (especially those using the same radio band) can wreak havoc on communications – a simple microwave oven can be an effective denial-of-service weapon.

2) Wireless networks are inherently less secure than wired ones (in theory).

In a wireless network, you are constantly broadcasting information to anyone who has a receiver. In a wired network, the data is directed along the cabling, and getting that data is much more difficult. Sure an attacker could use an inductive sensor to measure that data, but that is a more specialized attack, and you could always wrap the cable in a Faraday cage and bury it in concrete. With wireless you do not have any options.

3) Do not use WEP.

WEP (for Wired-Equivalent Privacy) was the original attempt at securing 802.11 networks. Unfortunately, the protocol was broken from almost the moment it was released and can now be compromised in seconds with freely-available software utilities. Think long and hard about connecting via WEP and if you do, make sure you are using additional authentication and encryption methods (see #6).

4) WPA-TKIP is probably a bad idea too. WPA was developed as a response to WEP being broken, and was designed to work with the same hardware. Unfortunately, it is showing signs that it might be time to drop it as well. It is not nearly as broken as WEP, but new attacks show that it is weak and will likely join WEP soon. Fortunately, WPA2 was developed as a complete (non-hardware-compatible) replacement. If you are using Wi-Fi encryption, opt for WPA2.

5) Wi-Fi encryption (WPA, WPA2, etc.) is not enough – you need TLS or similar.

A major caveat to using the Wi-Fi encryption protocols is that they only protect the data as it travels between the device and the Wi-Fi access point. Once the data reaches the access point, it is decrypted and passed along with no protection whatsoever. This might be okay on a private corporate network, but if your device is connected to the Internet you may as well just turn your WPA2 off – there is no security at that point. For this reason, it is advisable to use a higher-level security protocol like Transport Layer Security (TLS) or the Secure Shell (SSH) protocol. Those protocols encrypt everything from the device to the data's destination.

6) Do you need enterprise authentication?

Without authentication an attacker could pose as a legitimate receiver of information directly from your device. Authentication can be provided through requiring a simple password, pre-sharing a secret key, using the built-in authentication of TLS or similar protocols, or using full-blown enterprise authentication that requires a dedicated authentication server to guarantee the identity of all devices. The most common methods for authentication are EAP-TLS and PEAP.

7) How do you deploy and manage certificates and keys?

Whether you use an authentication method or just a pre-shared key, you need to manage your keys and/or digital certificates (certificates are used in enterprise authentication

and by TLS). Each device should have a unique certificate or key if you want the best security. You could distribute the same key to all of your devices, but then if one device is compromised, so is your entire system. It is a little harder to use the same certificate on every device since the authentication mechanism matches a certificate to the device's address. In any case, you will need to develop a way to distribute keys and certificates and update them as required.

8) Wi-Fi security protocols are big and slow.

If you are developing an embedded application with hundreds or thousands of units, the per-unit cost is likely an issue. This means you will likely need to scale back the performance of the device, but keep in mind that encryption is processor (and sometimes memory) intensive and the Wi-Fi security protocols are designed for the best security, with performance as a secondary requirement. When scoping the requirements for your hardware, be sure to include the requirements for the level of security you need.

9) What is your network infrastructure?

You should look at points of failure – for instance, if all of your devices communicate with a single access point, then that access point becomes a single point of failure for the entire system, and will be the likely target of any attacker. Adding some redundancy into your infrastructure can help alleviate this type of security concern.

10) Don't forget that the weakest component in any secure system is the user.

In any system, the user is the weakest link by a large margin. Passwords that are too simple, writing passwords on slips of paper, or just plain stupidity are often at the root of any successful attack. By acknowledging this fact, you can do a few things to mitigate the issues your users (including yourself) may cause. You should also consider who has access to the system and it is not a bad idea to limit what users can do. ■

Timothy Stapko is lead software engineer for Digi International - www.digi.com