Wireless Security for the Internet of Things

More and more Internet of Things products are appearing on the market, as start-ups and established companies combine to increase the momentum in this area. But a lot that I look at have only paid lip service to security. In this article I consider the current state of knowledge that has been applied to these emerging products and suggest the steps that designers need to take in the future.

Nick Hunn

WiFore Consulting

nick@wifore.com
+44 7768 890 148

Wireless Security for the Internet of Things

If you believe the futurologists, then the Internet of Things (IoT) is going to be the next big thing. Depending on who you listen to, by 2020 there will be up to 50 billion connected devices, an order of magnitude greater than the number of mobile phones. You can already see the start of that, whether it's smart meters, connected information signs, or the increasing number of fitness devices, like Fitbit and Nike's Fuel wristband. To get a better idea of what else may be emerging to make up that number, a good place to start is Kickstarter – the website for crowd-sourced funding. It shows that a significant number of potential start-ups are looking for money to produce a bewildering array of gateways and sensors.

It's great that there is so much innovation going on in this area. I've been trying to help it take off for almost two decades and at last I can convincingly say it's happening. But underneath the enthusiasm, I'm concerned that not enough attention is being given to security.

A few weeks ago, a speaker at a security conference in Australia talked about wireless attacks on pacemakers. Possibly because of the combined press frenzy around Superstorm Sandy, Obama's re-election and Jimmy Savile, that piece of information wasn't picked up by the mass media. At the same time, I've been playing with some of the latest consumer products that have come to market and found very little evidence of security. In fact, recent coverage in the technical press suggests there is a worrying feeling of complacency. I suspect that may be because wireless and end-to-end security is a new concept for many of the engineers designing IoT devices. But it is important that it makes its way onto the agenda, otherwise it may seriously impact the potential for growth.

What prompted me to write this article was an editorial piece in a copy of Connected World. I like Connected World magazine a lot – they've been pioneers in promoting M2M and do a neat job of straddling the line between trade journal and popular magazine. However, the Editor's answer to "Are my connected Devices Secure?" was "Overall the major players in M2M (machine-to-machine) are taking the necessary steps to minimise vulnerabilities across the value chain. To meet this need, players across the value chain team up to help make connected devices secure. Beginning with the device, OEMs (original equipment manufacturers) ensure partners adhere to security policies so products are not easily compromised".

That's a pretty vacuous statement which does little more than fill up a few column inches. It sounds suspiciously like "let's put on our Teflon gloves and try to juggle fish". The article goes on to say that "…with tech players creating more secure platforms, government bodies developing stricter regulations, and an overall awareness of the issues among users, security concerns can be minimised and even overcome". That's a worrying level of complacency to propagate. It may be the line that some M2M providers want to push, but it doesn't chime with my experience of

what's happening within the first flush of devices that are coming to market. Most of the companies I've come across in the IoT arena are designing at the bleeding edge of their comms experience. They're generally unaware of any government cybersecurity standards or stricter regulation and I'd argue that they don't need to be. Those standards are being designed for national infrastructure like the smart grid, which is not where most of the IoT is aimed at or is ever going to be. What Internet of Things designers need is practical, relevant advice.

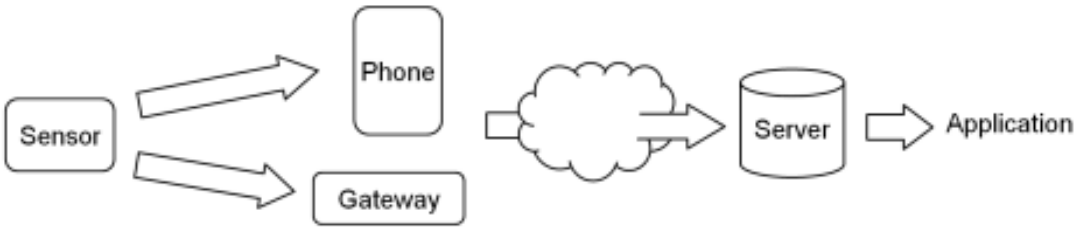
The other item I picked up on was a post on reverse engineering the Greengoose gateway. If you've not heard of Greengoose, have a look at their website and buy some to play with – they're neat. They are a good example of low cost innovation in connected sensors and apps for the home, which is everything that the IoT should be about. I suspect that they weren't designed with much of a requirement for security; after all, people who want to track their toothbrush or their toilet seat belong somewhere at the quantified-self end of the geek spectrum, for whom security and privacy have a somewhat different meaning than it does for most of the rest of the world. But the post illustrates that people will attempt to hack into these products. Not for any malicious reason, but because it's a challenge. It's the first thing I tried to do when I got my Greengoose kit. As I have with almost every other IoT product I've bought. But it illustrates the fact that wireless designers need to think carefully about what level of security a product needs.

The pacemaker and implanted defibrillator attack was the subject of a punchy session by Barnaby Jack at the Ruxcon Breakpoint security conference in Melbourne, Australia. The presentation made the point that it was too easy, pointing out that "there's no attempt to obfuscate or hide anything from a would-be attacker". That's an opinion in which they're not alone. A recent analysis of the security of AMI meters in the US concluded that not only were they vulnerable, but "most reverse engineering of the meter communication protocol required modest effort using off-the-shelf equipment".

Unfortunately that tends to be true for many wireless systems and products on the market today. The big standards – GSM, Bluetooth and Wi-Fi have learnt the need for security and have put a lot of effort into incorporating the tools to achieve it. However, they're often ignored by designers – a classic case being the "0000" PIN code which ruled amongst Bluetooth headsets and carkit manufacturers for many years. Even here, these security capable standards are limited to a fairly narrow bunch of high-volume consumer products. Almost all of the emerging IoT sensor-based devices which use wireless for control and monitoring use proprietary wireless protocols. And because most of them are put together and tested in isolation, they have little, if any security.

The rise of these connected devices, which is the vanguard of the Internet of Things, is really exciting. But it's mostly happening with low cost, proprietary wireless chips. The growth of products from new start-ups and Kickstarter projects are being fuelled by silicon vendors who are bringing highly integrated wireless processor chips to the market. The

tools that support these chips and reference designs make it very easy to get prototypes up and running and then take the resulting products to market. They're wonderful devices to design with, but they generally leave any security implementation to the designer. And wireless security is difficult. So where does an IoT designer start?



Most wireless sensor systems can be broken down into three distinct parts – the sensor, which generates the data; the gateway (which may be a mobile phone) that takes sensor data and transmits it over the WAN and the server/database which receives, stores and processes it. The traffic (at a high level) may be bidirectional, with control signals going back down to the sensor. And the WAN access may be an integral part of the sensor, as is the case where it contains a cellular modem. But in most cases it's not.

Where it is separate, there are typically two wireless links – the short range, local or personal area network and the wide area connection, which is generally either cellular or broadband. The chances are that any security implementation is different over both, and that there's not any end-to-end security. Most systems tend to be put together in a piecemeal "Lego" fashion, so security is at best only link wide. There are some vocal advocates of IP to the device, claiming that it plays to the end-to-end security model, but I'm still to be convinced that IP and low power wireless make sensible bedfellows. Which means that most real M2M and IoT implementations are likely to combine a number of different security schemes, without an overall end-to-end security model.

If security is important to you then one of the first things you need to do is to construct an end-to-end security model. Even if you think that it's not important for your application, it's still worth doing this, so that you can demonstrate why you didn't need to implement it. The principle here is that you need to think through what you are trying to protect and what the consequences of an attack may be. The severity of risk may not be where you think it is. Too often I've seen massive overkill in a sensor or gateway which then places secure data from multiple sources in an unsecured server.

I can't stress how important it is to do this at an early stage of the design process, as it affects the choice of protocols and chips. When it's done early on, it adds little cost or time to a project. Adding it as an afterthought can cripple the cost of a product or service, or at worst send it back to the drawing board.

As soon as you've decided what you want your product to do, and sketched out the overall architecture, sit down and produce what the industry calls an RMADS. I can't say I'm particularly proud to know that stands for a Risk Management and Accreditation Document Set, because it a big acronym for something that is essentially common sense. The philosophy behind it is to ask what is the consequence of data being lost, corrupted or injected at each stage? Each of these three possibilities is important to consider. And their relative importance will be different for different applications.

As with most big acronyms, you can pay a consultant a lot of money to generate your RMADS, but in most cases you can probably do a perfectly competent job yourself just by applying a little common sense and attempting to look at your product from the point of view of a user and of a hacker. And not just as a designer.

Common sense here really means thinking about fit for purpose. For many sensors around the home it may not matter if they can be overheard. It may matter more if someone can inject spurious packets, as that can lead to false alarms, the transmission of incorrect data that gets back to the server or the annoyance of something being turned on or off. All of which can reduce customer confidence in your product as it makes it look unreliable.

An associated point to consider is working out how to add new wireless devices to the network and stop rogue ones being attached. Pairing and authentication is one of the most difficult aspects of wireless, as ease of use and security come head to head. You also need to think about how to swap out defective devices without leaving vulnerabilities, which essentially means working out how to distribute link keys securely around your system.

At the gateway you need to consider how you ensure that the sensor data gets back to the server securely. Which generally means TLS, unless you have end to end security. That's not the way most IoT devices work today, as the community is promoting simplicity, open hardware, open APIs and simple POST messages. That doesn't means you can't design secure open systems, but you need to understand what level of security you're being offered and make sure you're happy with it. Once again, it's about understanding what you're implementing and whether it meets your needs. Remember that in most cases, commercial IoT products are only viable when the customer can trust the way their data is being handled. That's a very different scenario from people experimenting with Arduinos and open sensor projects. Each have their place, and can extend into each other's, but the inherent security levels of each should not be confused.

Often the biggest issue is at the server, where a lack of thought can expose the data. As we regularly see, even large companies who should know better don't handle their passwords and authentication robustly. And the bigger the target, the more interesting it gets to hackers. If you're designing a commercial system and you're lucky, it could catch the

zeitgeist and grow beyond your wildest dreams to become a significant part of those 50 billion devices. Some of the products being designed today are probably destined to do that. At which point any lack of security in their initial design will come back to haunt them and their investors. Which is another reason for getting it right at the beginning.

There's currently something of a fad in Government circles to set up new cybersecurity institutes all over the place – something which seems to be more a support mechanism for academics than a practical way of educating designers about security. And there seems to be a new one announced every week. These are not a panacea. In many ways, I think they're not only irrelevant, but a distraction to the more important security issue, which is practical help to get the security knowledge across to the current generation of IoT design pioneers. I'm planning to write about how to go about these basics in a future article. Until then, remember to think about the security of your device. And never assume that it comes automatically with the chips, stacks or protocols that you are using. Security is one of those things that needs thought and design. And if you don't take the time to consider it properly you could end up missing out on a great opportunity.

Nick Hunn

November 2012

Read more articles and posts at www.nickhunn.com