SCTE ISBE CABLE-TEC EXPO '16
SEPTEMBER 26-29 PHILADELPHIA

Society of Cable Telecommunications Engineers
International Society of Broadband Experts

Wireless Security Vulnerabilities & Solutions

Practical advice on finding and eliminating risk on wireless platforms

A Technical Paper prepared for SCTE/ISBE by

Christopher Kocks
Director, IoT Practice
Pure Integration, LLC.
Atlanta, GA
678-467-7458
Chris.kocks@pureintegration.com

Technical Contributions by
R. J. Brownlow
Security Researcher

Table of Contents

List of Figures

Introduction

1. Abstract

The rapid growth of Internet of Things (IoT) has dramatically expanded the number of wireless devices & platforms in the home, business, industry and around us. In addition to Wi-Fi we see dramatic growth of Bluetooth, ZigBee, and Z-Wave, and low power enabled devices. While connected devices have many advantages, they also provide greater attack surfaces and vulnerabilities for consumers and service providers. What are the security vulnerabilities to be aware of and how can solution architects design to reduce risk? Do IoT platforms provide adequate safe-guards? Which platforms provide the most reliable security? This paper will explore real examples of open platform security risks in home & public connected environments across several protocols. We will demonstrate several tools to illustrate how hackers compromise connected wireless devices & networks along with tactics to architect and prevent intrusion risk. Readers will see solutions that actually work to improve safety, cost savings, time savings, and convenience. The audience is anyone planning or designing an IoT solution based on an open platform. Product executives, solution architects, and security professionals will be interested in understanding security differences between open platforms and methods for designing proper safe-guards. Attendees will also see tools used to help test and validate these solutions.

2. Preview

Connectivity, home automation, energy conservation, security, health monitoring, business applications, agricultural and industrial uses remain driving factors of wireless communication. All of these have varying requirements in terms of bandwidth, cost, privacy, and installation. The development of Internet-connected technologies particularly require implementing IP solutions to harness energy savings and improve one's quality of life while staying safe from various security threats.

Several customized industry-standard-based networking protocols allow the fast growth and implementation of self-healing mesh networks, which are much more reliable network arrangements. Some of these networking protocols, including Z-Wave, ZigBee, and Bluetooth are based on the IEEE 802.15.4 protocol. They can enable cost-effective communication between devices with low latency and cheap installation costs. However, because there are several available protocols, security often suffers. Each protocol represents a new attack surface area for possible security flaws.

This paper highlights the importance of wireless security and cites some of the ways by which the lack of standards can place users at great risk. The focus is on internet-capable home & business appliances for which customized protocols were created. It also features attack scenarios based on home automation protocols based on IEEE 802.15.4.

This paper describes several plausible attacks that target smart home & business systems, using SDR (software defined radio) platforms. We will illustrate frameworks based on existing tools for practical, readily useable and hardware independent attacks. We will demonstrate multiple attack vectors that compromise the symmetric keys used to secure these networks, where both the originator and receiver must share this same key.

Additionally, this paper includes several tools for security researchers and professionals to consider to help them be most effective in eliminating cyber security risks on wireless platforms.

IoT Impacts

As the number of connected devices multiplies each year the security risks grow as well. Estimates from Cisco, Ericsson, IDC, ABI, Forrester, and Gartner all forecast between 25-50 billion connected devices by 2020. That translates to over 26 devices per person according to Intel. The potential economic impact estimated by these same firms is estimated between $2-$5 trillion in the same time period.

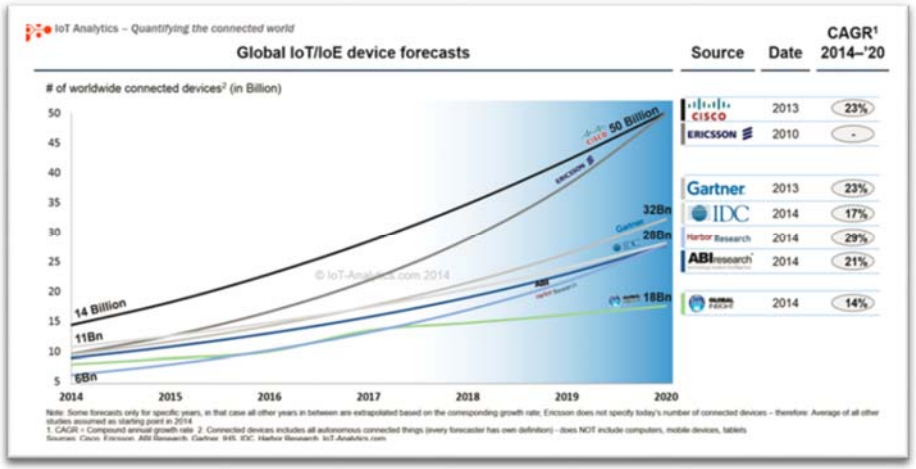

Figure 1 - Global IoT/IoE Device Forecasts

In parallel, the growth of IoT devices and platforms has contributed tremendously to the increase in cybersecurity issues according to BI Intelligence and leading agencies. Executives indicate that IoT is their single biggest threat and opportunity at the same time.

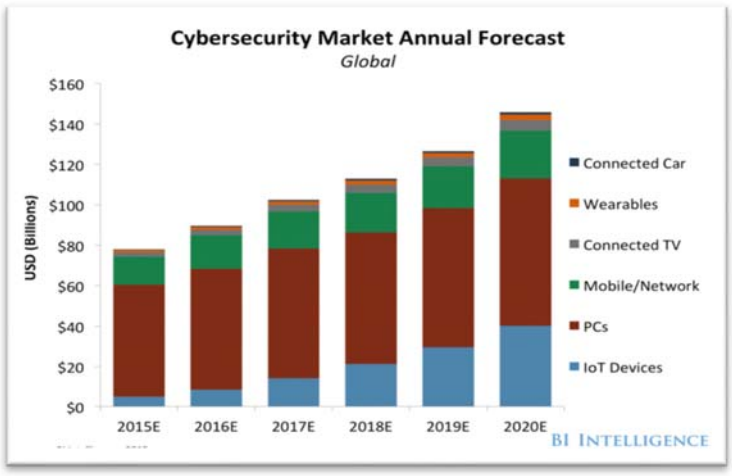

Figure 2 - Cybersecurity Market Annual Forecast

A ComputerWorld survey reveals that almost half of IT leaders said they will invest more next year in access control, intrusion prevention, filtering MAC addresses, identity management, and virus & malware protection.

With the explosion of connected devices & platforms comes multiple wireless technologies. These technologies range from near field, low power solutions to global persistent technologies. The IoT space includes the full spectrum when considering home, business, industrial, transportation, health, and global applications.

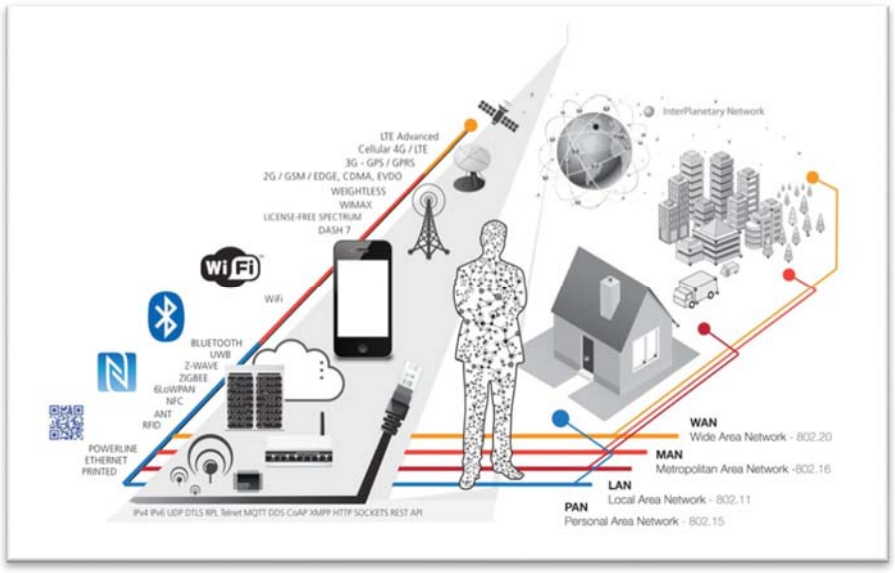

Figure 3 - IoT Wireless Technologies Spectrum by Postscapes.com

The security vulnerabilities illustrated for the selected technologies may also be applied to broader platforms in a similar manner. In the following sections this technical paper will examine exploits on just a couple of the most common retail applications available to consumers today. The potential vulnerability exposure is much greater than what can be presented in this publication.

Technology Brief: Z-Wave

As one of the leading wireless protocols in smart home automation, Z-Wave stands on the forefront of many consumers' first experiences with the IoT. Its prevalence has grown fast – 2014-2015 saw double-digit growth in sales of Z-Wave chips, eventually surpassing 35 million units. The ecosystem now has 250 manufacturers using the protocol in over 1,200 different devices.

Z-Wave primarily allows reliable transmission of short messages from a control unit to one or more nodes in a network. Its architecture comprises five main layers—the physical (PHY), Medium Access Control (MAC), transfer, routing, and application layers. It uses two types of device—controllers and slaves. Controllers poll or send commands to slaves, which either reply to or execute the controllers' commands.

Some homes can be fully controlled via a home automation system (e.g., sockets, TV sets, sound systems, lights, etc.). They may have started building their wireless personal area networks (WPANs) years ago so they would have various versions of Z-Wave chips (i.e., 200, 300, and 400 series).

As will be demonstrated below it's possible to sniff all of the traffic that flows in a WPAN. Anyone can learn to use professional tools like Wireshark, Kali Linux, and Freakduino Chibi Wireless Arduino-

compatible boards for an intended attack. Cybercriminals can easily view tutorials and buy tools to sniff WPAN traffic to discover a user's daily routine, what devices are in their home and how they are controlled.

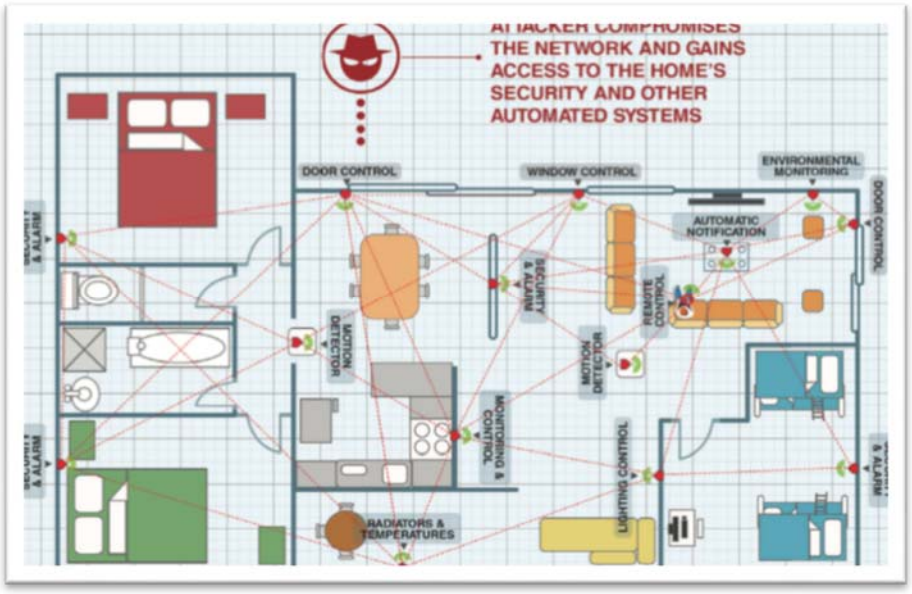

Figure 4 - Proposed Z-Wave attack blueprints

For one, knowing the day-to-day schedule of the owner of an automated home can let a thief know when the house is empty and easy to steal from. More tech-savvy thieves can also inject random commands to your WPAN, letting them turn connected devices on and off or change how these are set up. A well-examined attack scenario involves the remote sniffing of Z-wave packets and the injection of "unlock" packets on certain Z-Wave door lock products. For example, it would be possible to park across the street from a home or business and sniff packets when a person enters that building and replay those packets later to gain access. They can tinker with automated devices and/or appliances in homes and businesses, causing them to malfunction or potential harm.

Technology Brief: ZigBee

ZigBee is a low-data rate, low-power consumption, and low-cost wireless mesh networking protocol for automation and remote control applications. It comprises four basic layers—the PHY, MAC, network, and application layers— which provide additional security functionality.

Unlike Z-Wave, products based on ZigBee uses advanced encryption standard (AES) to encrypt messages. This makes it very hard to figure out possible attacks.

ZigBee has cryptographic support, which is enabled by default. Problems can only surface in the gateway between a WPAN and an internet protocol (IP) network. People normally trust ZigBee's security but forget about their IP networks. They forget that these need to be specially configured for safety.

If an attacker gains access to a gateway due to the use of a default or weak password, a misconfiguration, or lack of security, he can bypass ZigBee authentication. This will give him full access to the network, including security cameras. He can then see daily activities. He can also change the gateway configuration so connections will route to a fake Domain Name System (DNS) or proxy server. He can respond to all of the DNS queries and sniff all of the hypertext transfer protocol (HTTP) and secure (HTTPS) requests sent out. This will allow him to steal personally identifiable information (PII), including email and bank account credentials. With uninhibited access to the router, he can change your firewall settings and get direct access to any ZigBee-compliant device of his choosing.

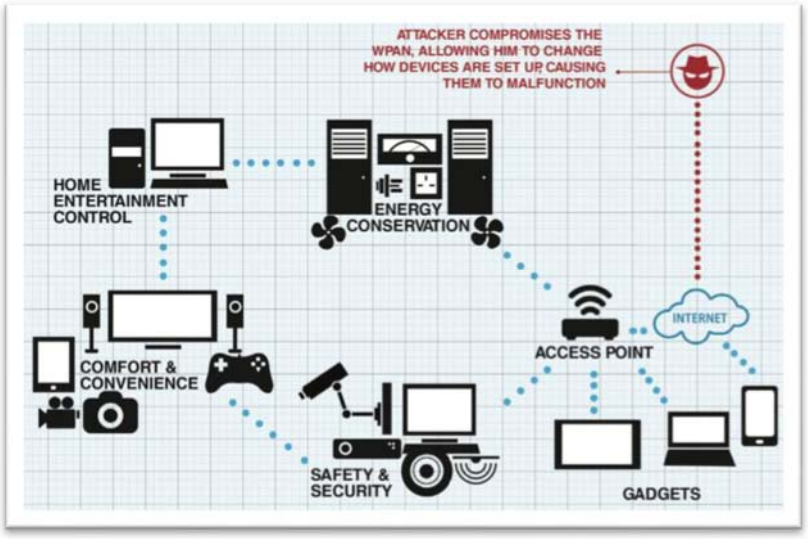

Figure 5 - Proposed ZigBee attack blueprints

Attack Vectors & Scenarios

The following section highlights 3 potential attack scenarios based on key manipulation. These scenarios represent attack vectors, or methods used by criminals to exploit IoT devices.

1. Executed Attack Scenario 1: Network Authentication Key Establishment

The use of pre-installed keys can be manipulated by the host/controller sending key manipulation data over the air (OTA) without actually dispensing keys. This works by each node on the network having a copy of several keys (32 being standard) with a key manipulation algorithm also being inherent to each node. The controller then sends the key manipulation data to each device with both the controller and node executing the command simultaneously. The final step is for the controller to check the value produced by the node against its own and authorize communication.

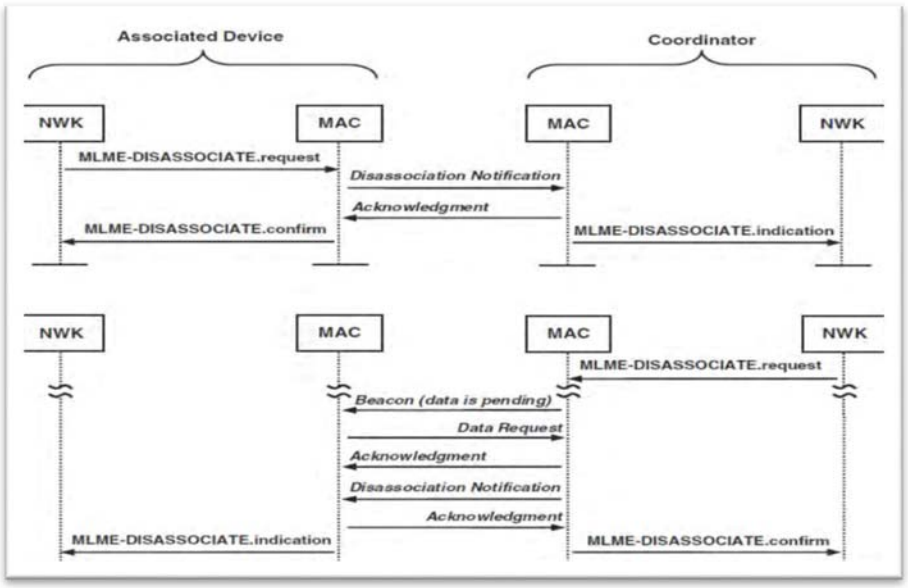

Figure 6 -Disassociation/ De-authorization attack diagram

This key establishment scheme can be easily manipulated by the use of a De-Authorization attack. The node being detached is programmed to accept the network key established by the gateway.

Once disconnected, the node will attempt to reestablish connection with the designated host but in many cases will default to the first host it finds.

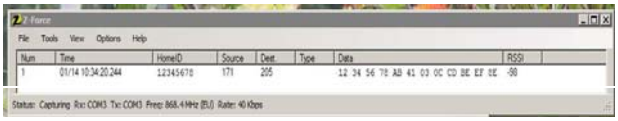

Figure 7 - Z-Wave node attempting to connect with a host/controller

As seen above, the node also sends the data associated with its last known host. Although encryption is in place, it's possible (but ineffective due to timing constraints and the use of low power transmissions) to simply record the key set message and extract the key from there. Due to the used encryption, it is also possible and feasible to calculate all necessary keys from captured packets

Once connected to the host (in this scenario, a laptop spoofing the controller) it will accept the key and any subsequent commands from its new host. In the screenshot below, the attacker uses this to send both a "SetKey" and "Unlock" command to the door.

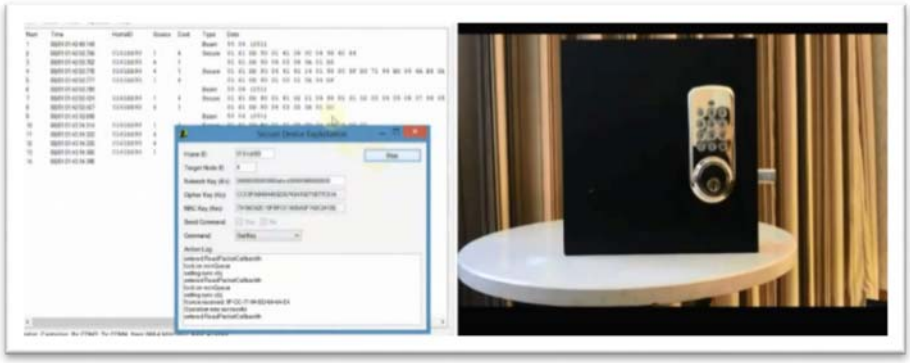

Figure 8 - Attacker sending commands to override the door

2. Executed Attack Scenario 2: Physical Key Extraction via Device Firmware

By default, keys are stored in every node of the network and can be extracted from the factory firmware by way of the debugger interface. In this exercise, a Z-Wave thermostat (manufacturer withheld), common of the shelf (COTS) flash programmer, factory development kit software tools and jumper wires are used to dump the firmware with a Serial to universal serial bus (USB) interface.

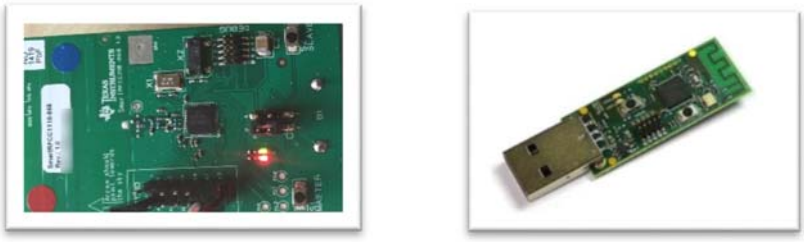

Figure 9 - Opened thermostat with debug access and USB debugging tool

Once accessed and dumped, the manufacturer's development kit can be used to decode the firmware code into plaintext as seen below.

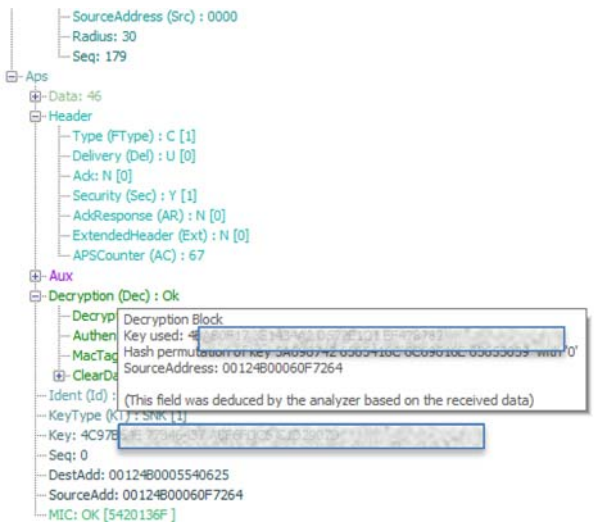

Figure 10 - Screenshot of the IDE displaying the encryption key in plain text

3. Executed Attack Scenario 3: OTA Key Transport Interception

In this authorization scheme, keys are transported directly to devices requesting to access the controller. The node sends a beacon broadcast to all devices in range (seen in red below), essentially looking for any network to join. The responding controller sends an acknowledgement and confirmation of availability (green). The node acknowledges receipt and requests a key to access the controller as a network resource (yellow). The controller responds with the network key and the node is added to the network (white-cropped out intentionally).

This entire transaction is sent in clear text and can easily be extracted by wireless sniffing methods.

Figure 11 - The network coordinator sends the key in plaintext to devices looking to join the network. Once acknowledged, all subsequent communication is immediately encrypted

Tools for IoT Security

Most Information Security Professionals are not typically trained to mitigate against hardware-based attacks. With that said, Information Security Professionals with experience in wireless security mitigation tactics often have building blocks to quickly get up to speed on the various attacks and defenses used with wireless radios.

The first thing that an Information Security Professional needs is a collection of hardware and software tools that will help assess the environment and give them the ability to develop an effective defense in depth.

One of the biggest challenges for researchers looking into wireless security is the high cost and complexity of obtaining software and application program interface (API) code from major semiconductor providers. Without the software that allows interaction with commercial radios, a researcher has limited ability to develop tools or other equipment that will run on multiple manufacturers' equipment. The purpose of this section is to offer several tool options for professionals to get up and running quickly, rather than providing an exhaustive comparison.

For spectrum analysis there are many expensive analyzers used by professionals for years. Fortunately, there are several newer options for portable or handheld analyzers scanning various ranges of spectrum. One example is the RF Explorer in the $100 - $500 range depending on the functionality required.

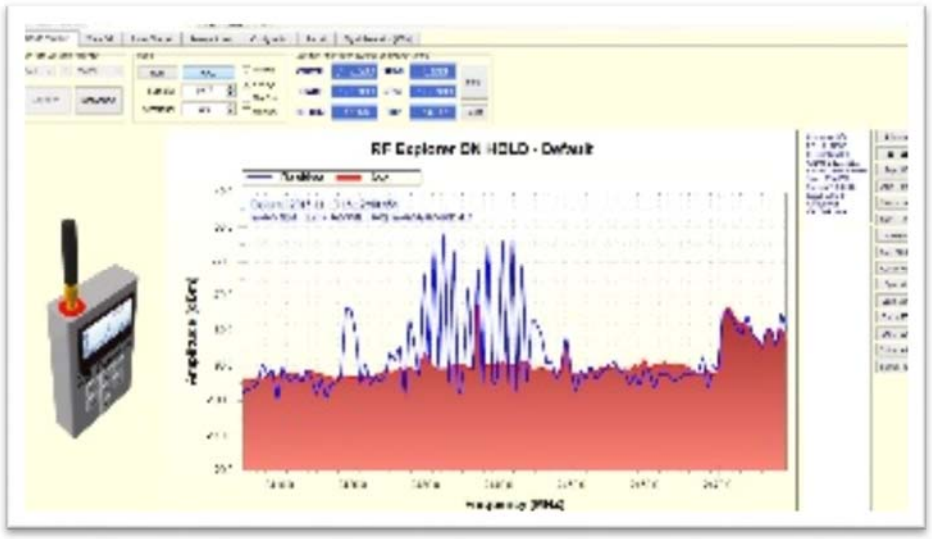

Figure 12 - RF Explorer Spectrum Analyzer

For packet sniffing, decryption, playback, there are almost too many choices in tools. However, many of these tools are often specific to a single device or protocol. Tools that cover multiple radios and protocols often run in the thousands of dollars, making them cost prohibitive for the average professional. The time and energy expended to wade through all the options can be enormous. The challenge is finding the best/broadest combination of radios and software at a reasonable cost & effort for a security professional. Fortunately, open source projects and even commercial solutions are moving in the right direction.

Let's begin with low cost solutions. One very effective toolset for key analysis is the KillerBee framework, which was created by Joshua Wright, a noted wireless security expert, and has been made freely available to everyone. KillerBee is really a suite of hardware and software tools that allow sophisticated interception, analysis, and even transmission of 802.15.4 packets. The software included in KillerBee is a collection of Python scripts that are easily modified and can be built upon to create even more capabilities and interaction with ZigBee radios. For Joshua's experiment, the recommended device of choice is the RZ Raven AVR, a $40 USB stick with monitoring and packet injection capabilities.

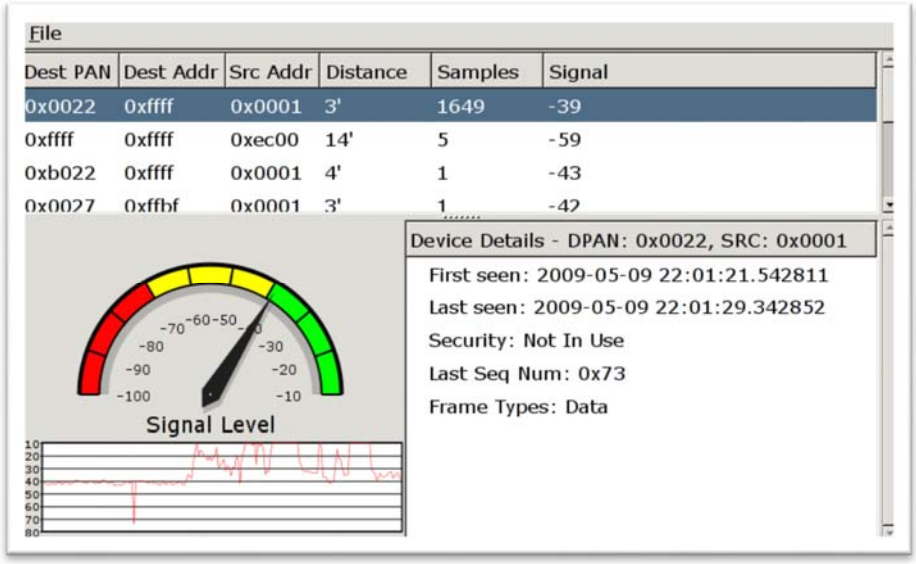

Figure 13 - KillerBee UI from Joshua Wright

For maker kit developers I assembled a RaspberryPi with multiple radio antennae running the Kali Linux implementation for penetration testing. The tool includes a 7" flat touchscreen display along with optional keyboard and mouse. Kali Linux is a Debian-derived Linux distribution utilized by researchers for penetration testing and forensics. The collection includes a wide variety of programs and utilities for application scanning, port scanning, packet analysis, penetration, password cracking, and attack management. The distribution can run on several other kits from BeagleBone, ARM, and other mobile devices. The mobility and powerful capability of this implementation is very appealing as it enables greater flexibility to get up close and personal to solve problems.

Figure 14 - Kali Linux on RaspberryPi2

Another very effective open source project is the Metasploit Framework originally developed by H.D. Moore. It is known widely as a portable tool for exploit development and vulnerability detection. The framework runs on both Unix and Windows based platforms.

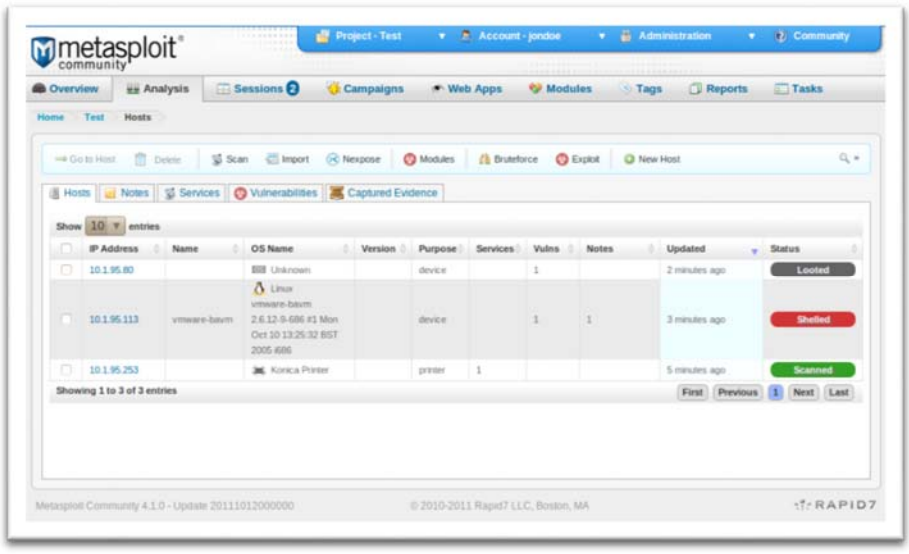

Figure 15 - Metasploit Framework

Texas Instruments (TI) offers several useful, low-cost wireless radios and software solutions. Each wireless radio can be purchased in a USB dongle form factor for around $50. The TI Sniffer and other software can be downloaded at no cost from the TI website. TI also has a nice LaunchPad kit for prototyping and connecting with various sensors and data platforms. For further analysis, the results can be imported into Wireshark, another common low budget solution.

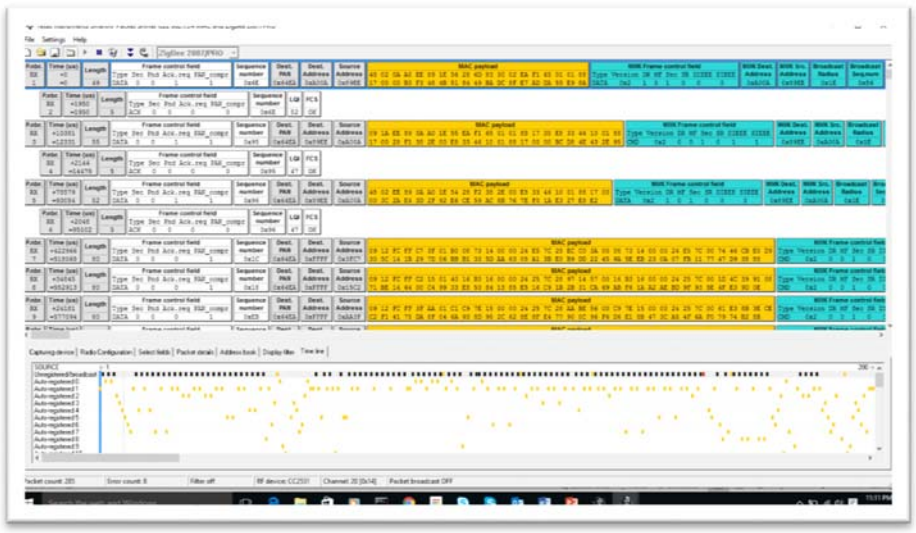

Figure 16 - TI Sniffer App Using TI CC2531 Dongle

A relatively affordable option originating from its Wi-Fi specialization is Wi-Spy inSSIDer software with the Mini adapter for under $250. This entry-level solution enables spectrum and channel analysis in the 2.4 GHZ range for multiple protocols. For a greater range of spectrum and analysis & reporting capabilities there are options in the $1k to $2k range.

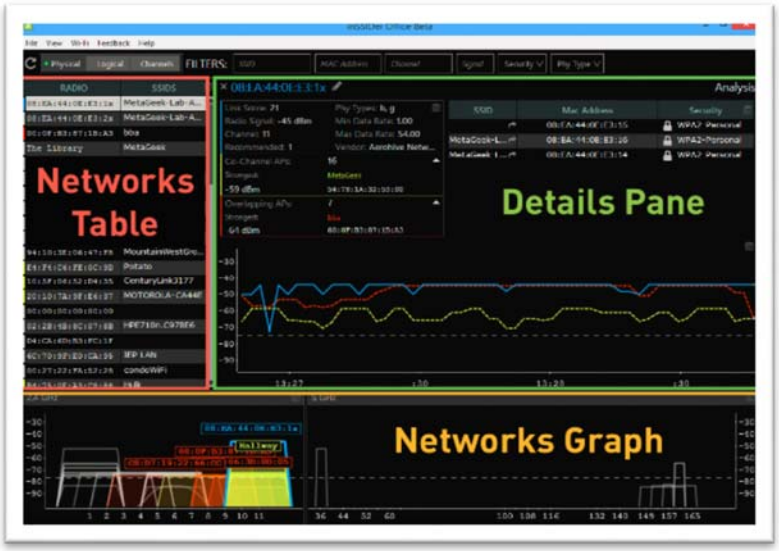

Figure 17 - Wi-Spy inSSIDer with Mini or DBx

For companies and professionals willing to spend a few thousand dollars, the SciLabs debugger device with the Ember Insight desktop software is a formidable combination used by communications companies to capture and debug packets in the lab and in the field. The cost is in the $3k-$5k range for the combination.

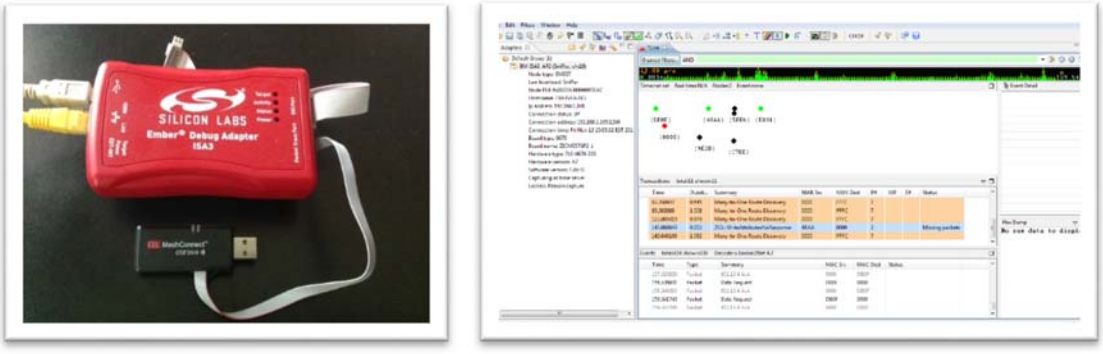

Figure 18 - SciLabs Debugger with Meshconnect Dongle and Ember Desktop

A robust commercial tool that provides extraordinary simplicity and out-of-the-box support for multiple dongles is software from Perytons. The software runs around $3k for the basic package with packet sniffing and decryption. The basic package includes real-time network mapping, performance stats, and complete interpretation of packet components and payloads. Researchers can upgrade with add-on modules for multi-channel, transaction play-back scripting, traffic generation, remote control, and advanced software development kit (SDK) capabilities depending upon the level of sophistication desired.

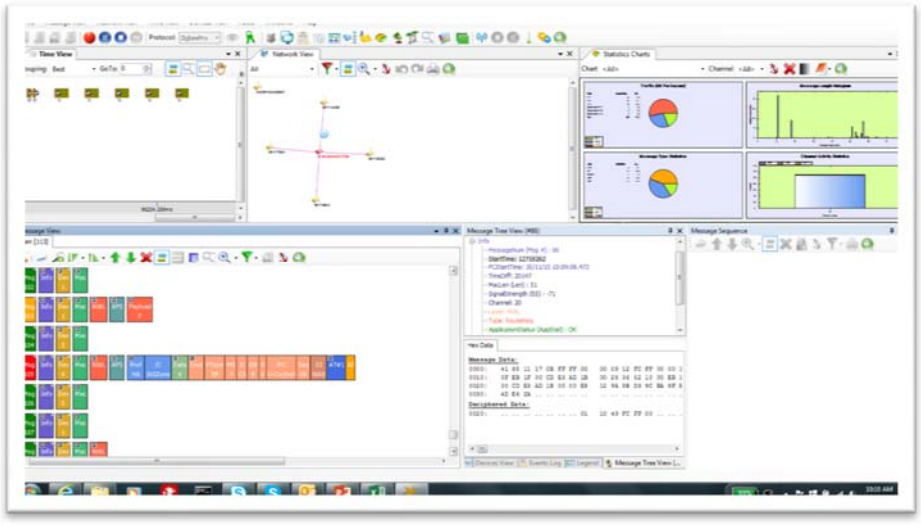

Figure 19 - Perytons Protocol Analyzer Software

Security professionals have a plethora of tools from which to choose. Although, it is even more imperative in wireless security to ensure professionals have the right tools for their unique jobs.

Combating Security Vulnerabilities

1. Security Strategy & Testing

This paper has highlighted the complexity of IoT solutions and the relative simplicity of exploiting security flaws in some devices and networks. How can organizations solve these security issues to protect their customers and constituents? Coordination between companies, standards bodies, device manufacturers, network operators, and software companies is essential to solving the end-to-end security challenge that currently does not exist in most platforms. We'll explore methods for these stakeholders to focus on solving security vulnerabilities together.

Companies offering or working with IoT solutions need to develop a comprehensive security strategy. The process begins with establishing a program if it doesn't already exist. The program may include collaboration with, and adoption of industry standards for IoT security. There are several industry standards bodies and groups that have published security standards, some including IoT specific models. Examples of standards organizations leading the way include GSMA, NIST, SANS, OWASP, ISO, and IEEE in addition to a few others.

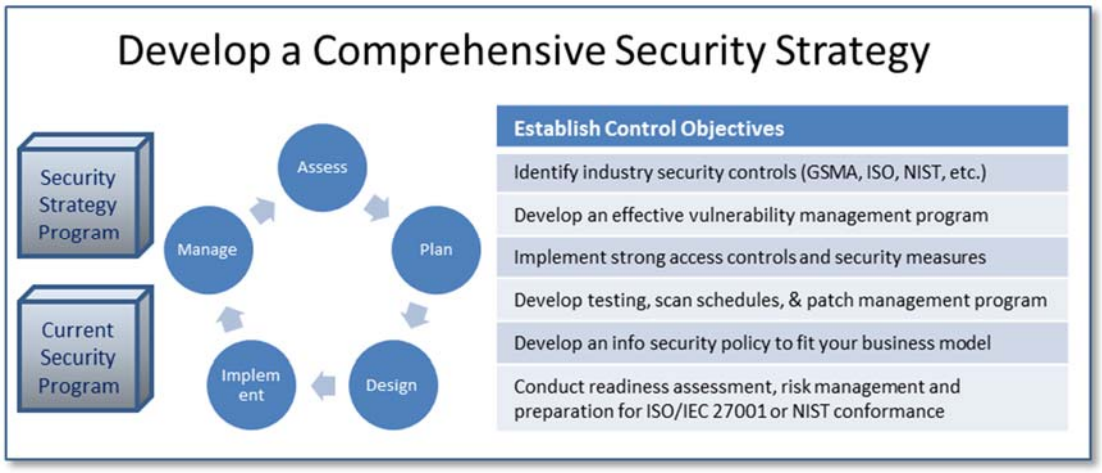

Figure 20 - Develop a Security Strategy

For companies who are heavy users, or resellers of IoT can mitigate risk by conducting vulnerability assessments and active penetration testing to expose attack surfaces and points of failure. The assessment and testing requires initial planning and mapping to prioritize highest risk areas and to plan for damage control from test results. Active testing often generates surprise results and sometimes potential system outages that must be anticipated. Finally, the risks should be re-prioritized based on impact and ability to mitigate real problems.

Figure 21 - Proactive Testing

2. Case Study on End-to-End Security

An example of how industries can implement end-to-end security solutions in IoT can be illustrated by an automotive scenario. Hacking into cars has been demonstrated for several years at DefCon and by independent researchers on many different vehicles. The following case study brief highlights how GSMA IoT guidelines can be implemented for comprehensive security safeguards in automotive and other industries.

The connected vehicle ecosystem is comprised of telematics systems that aggregate data, provide entertainment, and visualize diagnostics. A central computing system guides real-time decision making.

Embedded sensors guide drivers toward safe negotiation of road conditions. Wireless communication systems interact with nearby peers to relay safety critical metrics and alerts.

Common strategies used to attack automotive IoT technologies include exploiting weaknesses in telematics peer authentication, cryptographic tampering, compromising endpoint integrity in MCUs, blurred lines between and flaws in applications, and weaknesses in business logic. Savvy hackers confidently exploit network communications and physical endpoint device vulnerabilities embedded in the vehicle.

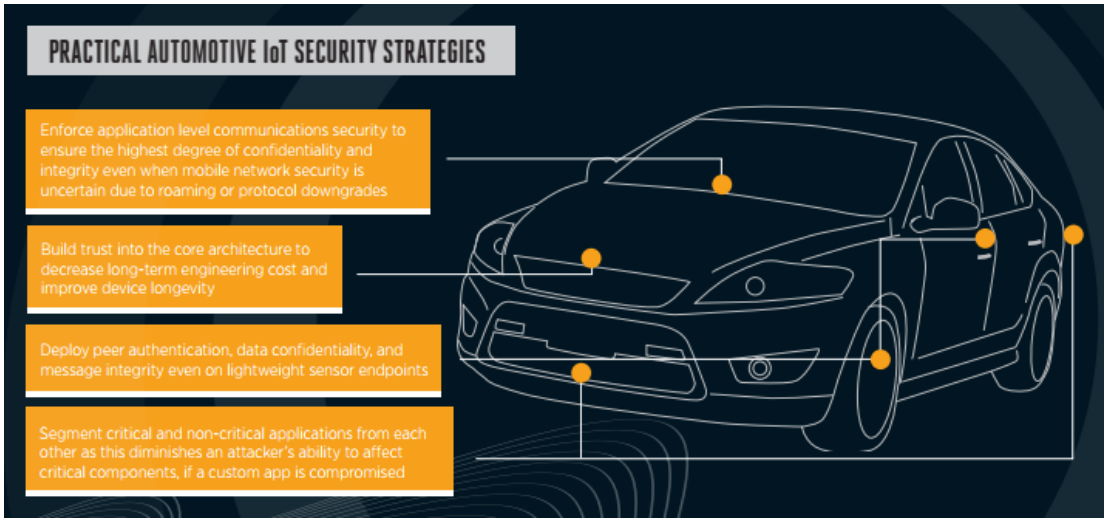

Figure 22 - Securing Automotive IoT (GSMA)

Solving these exploits is achievable with a thoughtful approach. GSMA's guidelines provide a good approach for protecting automotive IoT vulnerabilities. First, manufacturers should use a Trusted Computing Base with a collection of policies, procedures, and technologies that enforce the use and security of cryptographic and application-based tokens. Having a strong TCB is essential to a trust-worthy security solution. Next, network communications must be secured so that devices can authenticate and communicate with complete integrity. Networks must ensure that communications cannot be intercepted, altered, or impersonated. Application security is the next level of security required for trusted environments. The correct way to secure applications is by isolating them in jails, VMs, containers, or other abstraction that limits functionality and access to system devices and resources such as the CANbus. Finally, implementing device tamper resistance is an important strategy, even if to deter physical intrusion or render the attack non cost-effective. An example is to include a light-sensitive fuses or circuits that purge critical memory components when a device is opened improperly.

These guidelines are just a few developed by GSMA that can be applied to automotive and many other applications. IoT security issues can be solved but it requires building in security beginning with the architecture and early design while working aggressively with other solution partners. The ecosystem is still somewhat fragmented; although, standards bodies and solution providers are collaborating to deliver more secure and interoperable solutions.

Conclusion

The Internet of Things is, at the same time, providing extraordinary value while significantly increasing security vulnerabilities. There are many factors contributing to the risk, beyond the simple explosion in volume of usage. The IoT ecosystem is very complex, especially when platforms interoperate across different technologies at every layer of the stack (chips, devices, OS, network protocols, transport, applications, standards, and more). These complexities coupled with the cost of prevention are often cited as primary reasons for increasing security risks. While these factors have merit, a fundamental difference is that IoT is increasingly controlling devices that can cause great harm if exploited (vehicles, health devices, machinery, etc.). The potential negative impact can be tragic. Fortunately, security professionals do have weapons at their disposal and the economics are moving in their favor.

Component costs traditionally have been high but are becoming more economical. The IEEE 802.15.4 standard was created for use in residential and industrial markets. The more rapid proliferation of IoT in the industrial market has driven production costs down for necessary components (radio chipsets, microcontrollers, etc.). While this allows the residential segment to take advantage of the added cost benefits of proven technology, this also adds pressure to partners within the ecosystem to stay competitive. Unfortunately, the area that has suffered from cost cutting is device and end-to-end platform security. Encryption needs computational power, which requires hardware, which in turn adds cost. Partners will need to work together to spread costs across multiple organizations to develop a true ecosystem, where all are invested in security of the customer (and success of their products).

The industry needs more highly trained security professionals to deal with the rising risk. Experts and government officials echo a call for more qualified experts. As illustrated, tools and methodologies exist to aid professionals in IoT security prevention. One key to success will be to increase the number of security professionals trained on these tools and develop robust end-to-end security solutions to solve complex issues presented by disparate IoT ecosystems. Every industry goes through a maturation and optimization process. The IoT space will do the same over time. The challenge will be to get ahead of the curve before serious issues occur.

Abbreviations

AES	Advanced Encryption Standard
API	Application Program Interface
COTS	Common Off The Shelf
DNS	Domain Name Service
HTTP(S)	Hyper Text Transfer Protocol Secure
IoE	Internet of Everything
IoT	Internet of Things
IP	Internet Protocol
MAC	Media Access Control
OTA	Over The Air
PHY	Physical Layer
PII	Personally Identifiable Information
SDK	Software Development Kit
SDR	Software Defined Radio
TCB	Trusted Computing Base
TI	Texas Instruments
USB	Universal Serial Bus
WPAN	Wireless Personal Access Network

Bibliography & References

Picod, Jean-Michel; Lebrun, Arnaud; Demay, Jonathan-Christofer (2014). "Bringing Software Defined Radio to the Penetration Testing Community" (PDF). BlackHat USA. https://www.blackhat.com/docs/us-14/materials/us-14-Picod-Bringing-Software-Defined-Radio-To-The-Penetration-Testing-Community-WP.pdf

Business Insider: Hacking IoT http://www.businessinsider.com/iot-cyber-security-hacking-problems-internet-of-things-2016-3

AT&T Cybersecurity Insights https://www.business.att.com/cybersecurity/

CISCO/Forrester IoT Security Survey http://www.cisco.com/c/dam/en_us/solutions/trends/iot/vital-element.pdf

"Member Companies of the Z-Wave Alliance - Z-Wave Alliance". http://z-wavealliance.org/z-wave_alliance_member_companies/

Z-Wave Products http://products.z-wavealliance.org/

Z-Wave Certification http://z-wavealliance.org/sigma-designs-z-wave-alliance-introduce-new-z-wave-plus-certification-program/

"xPL Monkey - Home Automation - Z-Wave". www.xplmonkey.com/zwave.html

FreakZ open source Zigbee project http://www.sourceforge.net/projects/freakz

Zigbee/802.15.4 chip comparison http://freaklabs.org/index.php/Articles/Zigbee/Zigbee-Chip-Comparison.html

ZigBee wants to be the Bluetooth of the internet of things. Too bad everyone hates it. http://gigaom.com/2013/08/30/zigbee-wants-to-be-the-bluetooth-of-the-internet-of-things-too-bad-everyone-hates-it/

"KillerBee: Practical ZigBee Exploitation Framework", Joshua Wright, http://www.willhackforsushi.com/presentations/toorcon11-wright.pdf

GSMA IoT Security Guidelines http://www.gsma.com/connectedliving/iot-security-guidelines

SANS: Securing IoT Survey https://www.sans.org/reading-room/whitepapers/analyst/securing-internet-things-survey-34785

SANS https://www.sans.org/

OWASP https://www.owasp.org/index.php/Main_Page

Texas Instruments http://www.ti.com/

Silicon Labs http://www.silabs.com/Pages/default.aspx

California Eastern Labs http://www.cel.com/

Perytons http://www.perytons.com/

Wireshark https://www.wireshark.org/

Linux Kali https://www.kali.org/

Metasploit http://www.metasploit.com/

KillerBee Repository https://code.google.com/archive/p/killerbee/

Airsnort http://www.shmoo.com/

Netstumbler (maps networks with GPS) http://www.netstumbler.com/

Metageek Wi-Spy http://www.metageek.com

RF Explorer Spectrum Analyzer http://rfexplorer.com/

