Defending IoT Infrastructures with the Raspberry Pi

Monitoring and Detecting Nefarious Behavior in Real Time

Chet Hosmer

Apress®

Defending IoT Infrastructures with the Raspberry Pi

Monitoring and Detecting Nefarious Behavior in Real Time

Chet Hosmer

Apress®

Defending IoT Infrastructures with the Raspberry Pi: Monitoring and Detecting Nefarious Behavior in Real Time

Chet Hosmer
Longs, South Carolina, USA

ISBN-13 (pbk): 978-1-4842-3699-4 ISBN-13 (electronic): 978-1-4842-3700-7
https://doi.org/10.1007/978-1-4842-3700-7

Library of Congress Control Number: 2018949207

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Managing Director, Apress Media LLC: Welmoed Spahr
Acquisitions Editor: Susan McDermott
Development Editor: Laura Berendson
Coordinating Editor: Rita Fernando

Cover designed by eStudioCalamar

Cover image designed by Freepik (www.freepik.com)

Distributed to the book trade worldwide by Springer Science+Business Media New York, 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a **Delaware** corporation.

For information on translations, please e-mail rights@apress.com, or visit http://www.apress.com/rights-permissions.

Apress titles may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Print and eBook Bulk Sales web page at http://www.apress.com/bulk-sales.

Any source code or other supplementary material referenced by the author in this book is available to readers on GitHub via the book's product page, located at www.apress.com/9781484236994. For more detailed information, please visit http://www.apress.com/source-code.

Printed on acid-free paper

To my wife Janet; your love and guidance make the journey complete.

Table of Contents

About the Author

Chet Hosmer is the Founder of Python Forensics, Inc., a nonprofit organization focused on the collaborative development of open source investigative technologies using the Python programming language. Chet has been researching and developing technology and training surrounding forensics, digital investigation, and steganography for over two decades. He has made numerous appearances to discuss emerging cyberthreats, including National Public Radio's *Kojo Nnamdi Show*, ABC's *Primetime Thursday*, NHK Japan, TechTV's CyberCrime and ABC News Australia. He has also been a frequent contributor to technical and news stories relating to cybersecurity and forensics and has been interviewed and quoted by IEEE, *The New York Times*, *The Washington Post*, *Government Computer News*, Salon.com, and *Wired Magazine*.

Chet has authored five books within the cybersecurity domain, ranging from data hiding to forensics.

Chet serves as a visiting professor at Utica College in the Cybersecurity Graduate Program. He is also an adjunct faculty member at Champlain College in the Digital Forensic Science Program Masters Program.

Chet delivers keynote and plenary talks on various cybersecurity-related topics around the world each year.

About the Technical Reviewer

Michael T. Raggo
Chief Security Officer, 802 Secure (CISSP, NSA-IAM, ACE, CSI) has over 20 years of security research experience. His current focus is wireless IoT threats impacting the enterprise. Michael is the author of *Mobile Data Loss: Threats and Countermeasures* and *Data Hiding: Exposing Concealed Data in Multimedia, Operating Systems, Mobile Devices and Network Protocols* for Syngress Books, and contributing author for *Information Security: The Complete Reference* (2nd edition). A former security trainer, Michael has briefed international defense agencies including the FBI and Pentagon, is a participating member of FSISAC/BITS and PCI, and is a frequent presenter at security conferences, including Black Hat, DEF CON, Gartner, RSA, DoD Cyber Crime, OWASP, HackCon, and SANS.

Acknowledgments

A special thanks to Mike Raggo for his insight and encouragement throughout this process. Thank you for championing this project from the beginning and testing every version of the Pi sensor along the way. Your guidance, your friendship, and your quest to improve security and safety are inspiring.

Thanks to Carlton Jeffcoat and Cameron Covington at WetStone for deploying the sensor in live environments to passively map potential vulnerabilities and for providing insights to make the sensor even better.

Thanks to the Utica College cybersecurity graduate students who have experimented with the earliest to the final versions of PiSensor and have provided excellent feedback.

Thanks to Rita Fernando, Laura Berendson, Susan McDermott, and the whole team at Apress for your incredible patience throughout this process and for your constant encouragement.

Introduction

The Internet of Things (IoT) and industrial control systems (ICS) require special attention from a cybersecurity point of view. This is based on the well-known and -documented fact that the protocols and implementations have vulnerabilities that when exploited can produce considerable damage and provide an avenue for the exfiltration of data.

In addition, when examining these environments due to the dynamic nature and/or critical infrastructure implications, active scanning or probing of these environments is either discouraged or ineffective. Thus, passive monitoring of these environments offers insights into the behavior of these devices and the networks in which they operate. One of the core issues is the placement of the monitoring devices to provide visibility and coverage from both the wired and wireless points of view. There are vendor solutions that are offered today that rely on expensive hardware and software solutions that may lack flexibility.

Using a Raspberry Pi and open source Python software to passively monitor, detect, baseline, and provide insight into these behaviors has been called "crazy" by some. However, as you will see, the Raspberry Pi itself, with its multicore processor and integrated wired and wireless network components, provides the basic underpinnings necessary for a lightweight IoT/ICS sensor for less than $50.00. Couple that with an open source extensible Python software solution that dynamically reduces and records the most pertinent observations, and you have a low-cost, flexible, and nimble PiSensor for IoT and ICS environments.

CHAPTER 1

IoT Vulnerabilities

The Internet of Things (IoT) is a network of processing devices with unique identities that can connect to and transfer data over a network without requiring direct human interaction (see Figure 1-1). In many cases this makes the devices themselves autonomous or semiautonomous. They can be controlled, managed, and programmed to follow specific rules of engagement.

Interconnected autonomous devices
Limited or No Human Intervention
Supporting a variety of applications and platforms

***Figure 1-1.** IoT interconnected*

The breadth of devices that currently exist as of this writing include the following:

- Health and Fitness Monitoring
- Manufacturing Systems
- Energy Metering

C. Hosmer, *Defending IoT Infrastructures with the Raspberry Pi*,
https://doi.org/10.1007/978-1-4842-3700-7_1

- Hospital and Patient Care
- Smart Appliances and Lighting
- Enhanced Surveillance Systems
- Entertainment
- Home Automation and Security
- Multifunction Wearable Technologies
- Automotive
- Tracking Systems
- Personal Communications
- Along with new categories emerging every day

Note The focus of this book and the accompanying source code is to observe, learn, model, and detect aberrant behavior of IoT devices using the Raspberry Pi as a sensor.

Why Is IoT Vulnerable?

When considering vulnerabilities of IoT devices and networks, we must first define the overall attack surface. If you believe Gartner's prediction (Gartner Research, 2017) that 25.1 billion IoT endpoints will exist by the year 2021[1], then this would certainly define a large attack surface. Many of these devices are also interconnected and operating across boundaries of consumers, business, industry, and government, without geographic restrictions.

[1] Gartner: *Forecast: Internet of Things – Endpoints and Associated Services, Worldwide 2017*. www.gartner.com/doc/3840665/forecast-internet-things--endpoints.

Deployment options for IoT differ widely depending upon their application, industry, and defined use. However, we can generally classify IoT deployments in one of three ways: device to device, device to cloud, or device to gateway, as shown in Figures 1-2, 1-3, and 1-4.

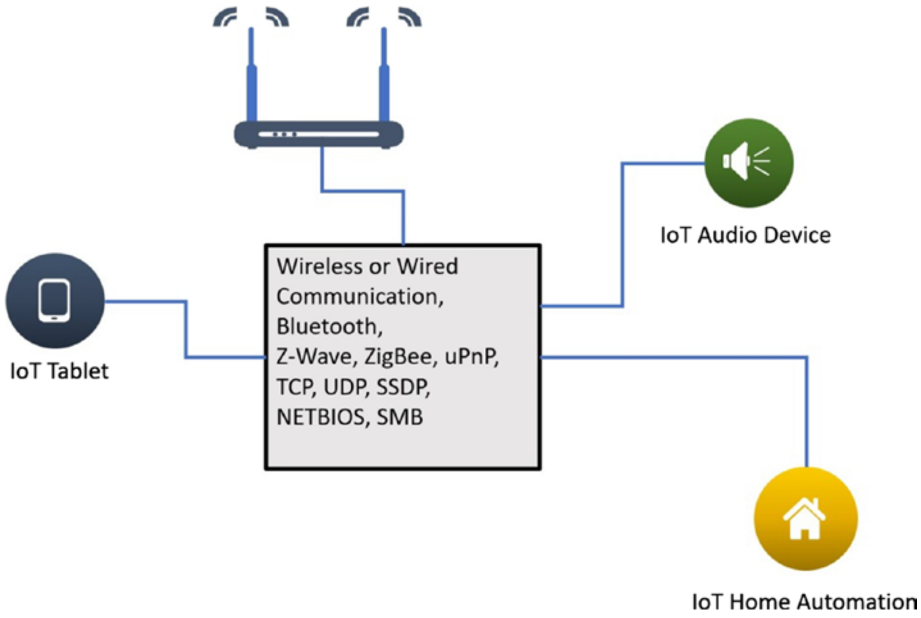

Wireless or Wired Communication, Bluetooth, Z-Wave, ZigBee, uPnP, TCP, UDP, SSDP, NETBIOS, SMB
IoT Tablet
IoT Audio Device
IoT Home Automation

Figure 1-2. *Device-to-device communication model*

Device-to-Device Communication

This simple model depicts devices that directly discover, connect, and communicate using the locally available networks. The communication can be through traditional TCP (Transaction Control Protocol)/UDP (User Datagram Protocol)/IP (Internet Protocol) networks; however, in many cases, they communicate over low-power or wireless networks such as Bluetooth, Z-Wave, ZigBee, and Universal Plug and Play (uPnP).

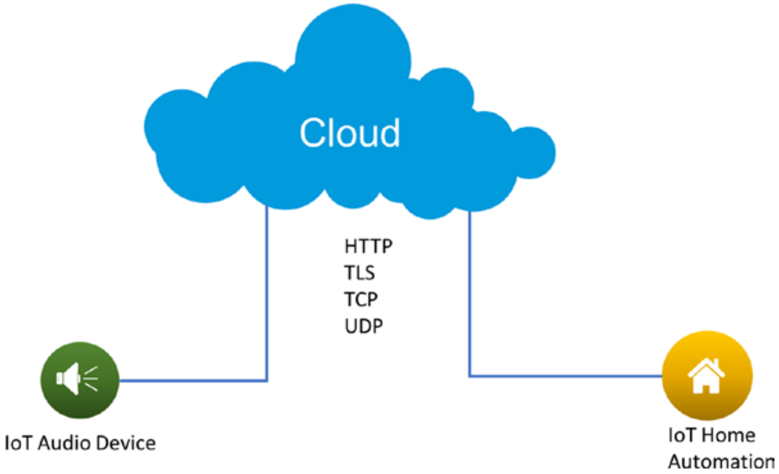


Figure 1-3. *Device-to-cloud communication*

Device-to-Cloud Communications

IoT devices using this method connect directly to an Internet-based cloud service to exchange data and control messages. This method typically utilizes traditional protocols such as TCP, UDP, HTTP(S), and TLS (Transport Layer Security) for security-based exchanges.

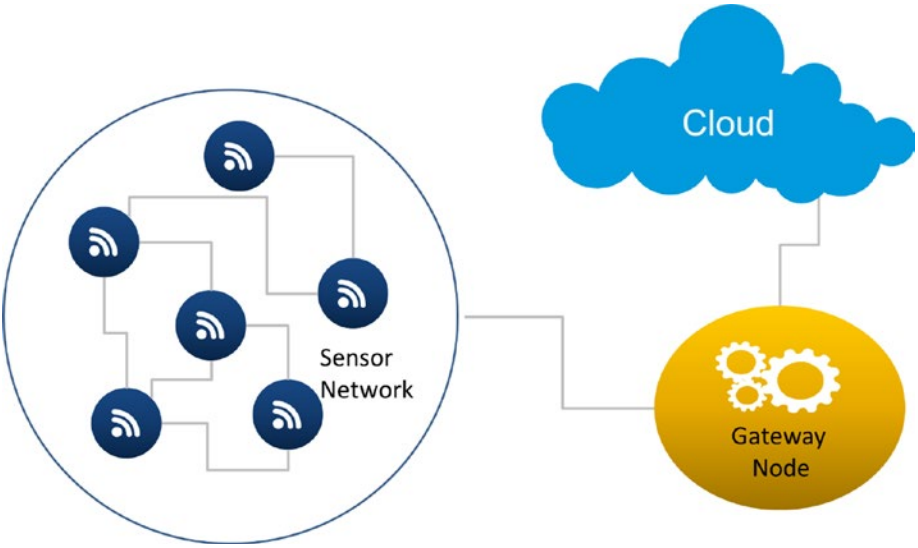


Figure 1-4. *Device-to-gateway framework*

Device-to-Gateway Sensor Network Communications

Utilizing this method, sensors discover and communicate with other sensors and coordinate information through gateways. The gateway, in turn, communicates information with other sensor networks and typically with the cloud.

At first glance, these connection and communications models don't look that different from more traditional distributed computing environments. However, many of the underlying protocols and methods of deployment are dissimilar from traditional environments and require closer examination. From a cybersecurity point of view, we still must consider and examine these environments using proven principles. At the heart, of course, is the CIA triad as shown in Figure 1-5.

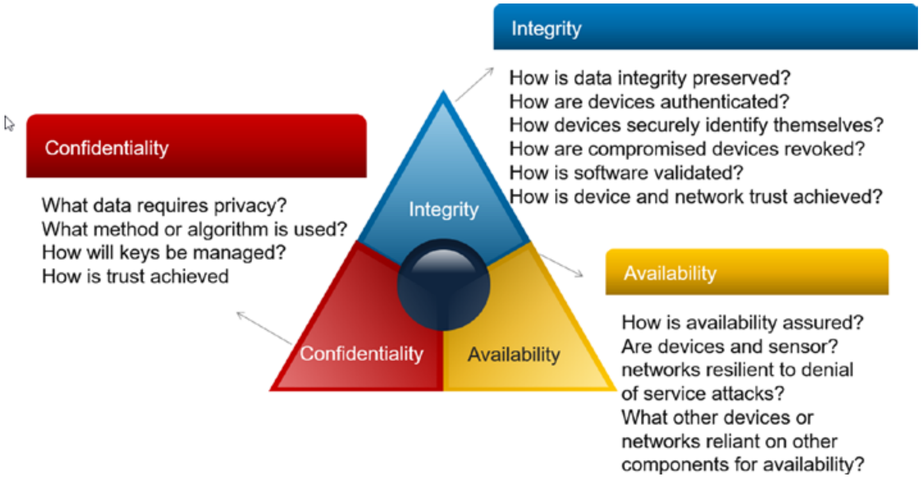
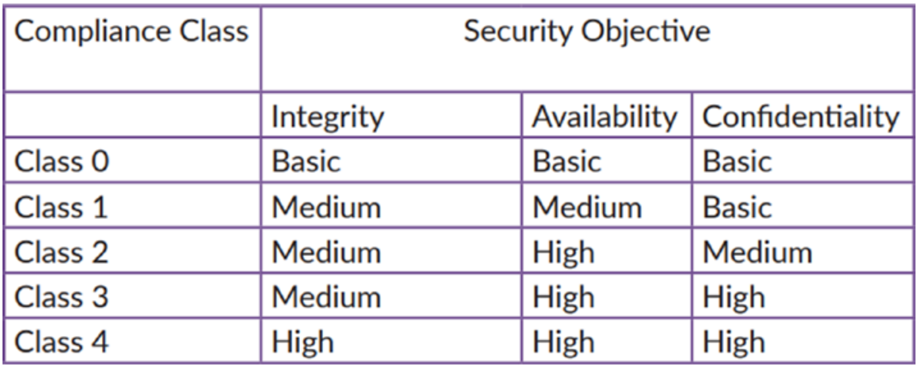

Integrity
How is data integrity preserved?
How are devices authenticated?
How devices securely identify themselves?
How are compromised devices revoked?
How is software validated?
How is device and network trust achieved?
Confidentiality
What data requires privacy?
What method or algorithm is used?
How will keys be managed?
How is trust achieved
Integrity
Confidentiality
Availability
Availability
How is availability assured?
Are devices and sensor?
networks resilient to denial
of service attacks?
What other devices or
networks reliant on other
components for availability?

Figure 1-5. *CIA triad*

The IoT Security Foundation published the *IoT Security Compliance Framework* in 2016 to help promote contemporary best practices in IoT security. As part of the framework, they applied the CIA triad to different classes of IoT devices as shown in Figure 1-6. They defined five specific classes of IoT devices along with the security requirements of each.

- **Class 0**: Compromise of data would cause little or no impact.
- **Class 1**: Compromise of data would cause limited impact.

- **Class 2**: Devices must be resilient to attack on availability that would have significant impact.
- **Class 3**: Devices must both be resilient to attack and protect sensitive data.
- **Class 4**: Devices must be resilient to attack, preserve integrity of operation, and protect sensitive data. Any resulting breach would cause serious impact and potentially cause injury.



Compliance Class	Security Objective		
	Integrity	Availability	Confidentiality
Class 0	Basic	Basic	Basic
Class 1	Medium	Medium	Basic
Class 2	Medium	High	Medium
Class 3	Medium	High	High
Class 4	High	High	High

Figure 1-6. *Compliance classification security objectives*

Interpreting the security objectives at each level are defined here in Table 1-1.

Table 1-1. *Interpreting the Security Levels*

Category	Level	Requirements
Integrity	Basic	IoT devices resist low-level threat sources that have very little capability
	Medium	IoT devices resist medium-level threat sources that have minimal focused capability
	High	IoT devices must resist substantial-level threat sources
Confidentiality	Basic	IoT devices processing public information
	Medium	IoT devices protect against disclosure of low-value personally identifiable information
	High	IoT devices process very sensitive information and must protect against any disclosure
Availability	Basic	IoT device lack of availability would cause only minor disruption
	Medium	IoT devices should possess some availability defenses against the most common attacks
	High	IoT devices must anticipate determined availability attacks and take significant measures to overcome them