Advanced Systems Security: Internet of Things

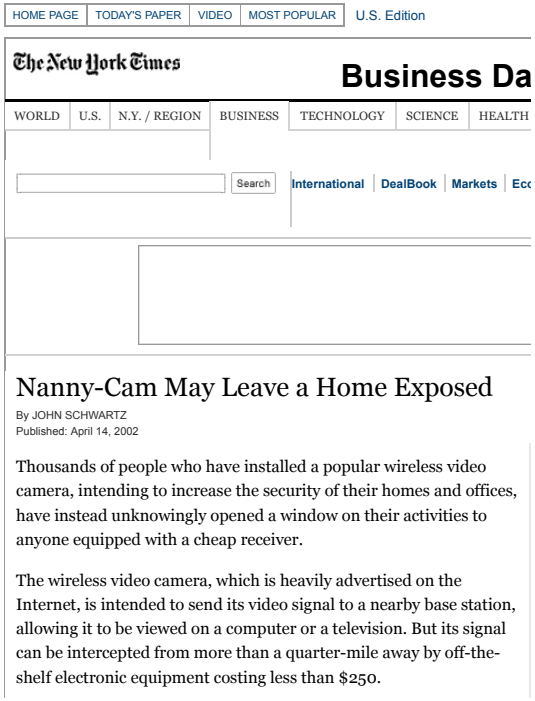
Trent Jaeger
Systems and Internet Infrastructure Security (SIIS) Lab
Penn State University

Connecting Things

- Product group works for years on a standalone appliance
  - Software development
  - System configuration
  - System maintenance (testing)
- Then, the company decides to connect the product to the Internet
  - To broaden utility and uses
- Then what happens?

Things Connected to Internet

- Cameras (Nanny Cams), 2002
  - Cameras employ wireless communication to convey data, but the wireless signal is not encrypted
  - Wireless not necessary for past applications (video recording)



HOME PAGE | TODAY'S PAPER | VIDEO | MOST POPULAR | U.S. Edition

The New York Times — Business Da

WORLD | U.S. | N.Y. / REGION | BUSINESS | TECHNOLOGY | SCIENCE | HEALTH

Search | International | DealBook | Markets | Ecc

Nanny-Cam May Leave a Home Exposed

By JOHN SCHWARTZ
Published: April 14, 2002

Thousands of people who have installed a popular wireless video camera, intending to increase the security of their homes and offices, have instead unknowingly opened a window on their activities to anyone equipped with a cheap receiver.

The wireless video camera, which is heavily advertised on the Internet, is intended to send its video signal to a nearby base station, allowing it to be viewed on a computer or a television. But its signal can be intercepted from more than a quarter-mile away by off-the-shelf electronic equipment costing less than $250.

Things Connected to Internet

- Medical Devices (Pacemakers), 2008
  - Remote adversary can cause data leakage to unauthenticated device and maliciously reprogram the ICD to change its operation
  - Slashdot (10/20/2015): Why aren't there better cybersecurity regulations for medical devices?

Pacemakers and Implantable Cardiac Defibrillators: Software Radio Attacks and Zero-Power Defenses

Daniel Halperin† University of Washington

Thomas S. Heydt-Benjamin† University of Massachusetts Amherst

Benjamin Ransford† University of Massachusetts Amherst

Shane S. Clark University of Massachusetts Amherst

Benessa Defend University of Massachusetts Amherst

Will Morgan University of Massachusetts Amherst

Kevin Fu, PhD* University of Massachusetts Amherst

Tadayoshi Kohno, PhD* University of Washington

William H. Maisel, MD, MPH* BIDMC and Harvard Medical School

***Abstract*—Our study analyzes the security and privacy properties of an implantable cardioverter defibrillator (ICD). Introduced to the U.S. market in 2003, this model of ICD includes pacemaker technology and is designed to communicate wirelessly with a nearby external programmer in the 175 kHz frequency range. After partially reverse-engineering the ICD's communications protocol with an oscilloscope and a software radio, we implemented several software radio-based attacks that could compromise patient safety and patient privacy. Motivated by our desire to improve patient safety, and mindful of conventional trade-offs between security and power consumption for resource-constrained devices, we introduce three new zero-power defenses based on RF power harvesting. Two of these defenses are human-centric, bringing patients into the loop with respect to the security and privacy of their implantable medical devices (IMDs). Our contributions provide a scientific baseline for understanding the potential security and privacy risks of current and future IMDs, and introduce human-perceptible and zero-power mitigation techniques that address those risks. To the best of our knowledge, this paper is the first in our community to use general-purpose software radios to analyze and attack previously unknown radio communications protocols.**

this event to a health care practitioner who uses a *commercial device programmer*[1] with wireless capabilities to extract data from the ICD or modify its settings without surgery. Between 1990 and 2002, over 2.6 million pacemakers and ICDs were implanted in patients in the United States [19]; clinical trials have shown that these devices significantly improve survival rates in certain populations [18]. Other research has discussed potential security and privacy risks of IMDs [1], [10], but we are unaware of any rigorous public investigation into the observable characteristics of a real commercial device. Without such a study, it is impossible for the research community to assess or address the security and privacy properties of past, current, and future devices. We address that gap in this paper and, based on our findings, propose and implement several prototype attack-mitigation techniques.

Our investigation was motivated by an interdisciplinary study of medical device safety and security, and relied on a diverse team of area specialists. Team members from the security and privacy community have formal training

Things Connected to Internet

- Smart Devices (Smart Grid), 2010
  - Rather than people reading meters (no longer manual – read like nanny cams ironically), have meters become part of an advanced metering infrastructure
  - Thefts enabled by password extraction, eavesdropping, meter spoofing, etc.

Energy Theft in the Advanced Metering Infrastructure

Stephen McLaughlin, Dmitry Podkuiko, and Patrick McDaniel

Systems and Internet Infrastructure Security Laboratory (SIIS)
Pennsylvania State University, University Park, PA
{smclaugh,podkuiko,mcdaniel}@cse.psu.edu

Abstract. Global energy generation and delivery systems are transitioning to a new computerized "smart grid". One of the principle components of the smart grid is an advanced metering infrastructure (AMI). AMI replaces the analog meters with computerized systems that report usage over digital communication interfaces, e.g., phone lines. However, with this infrastructure comes new risk. In this paper, we consider adversary means of defrauding the electrical grid by manipulating AMI systems. We document the methods adversaries will use to attempt to manipulate energy usage data, and validate the viability of these attacks by performing penetration testing on commodity devices. Through these activities, we demonstrate that not only is theft still possible in AMI systems, but that current AMI devices introduce a myriad of new vectors for achieving it.

Things Connected to Internet

- Complex Distributed Computer Systems (Automobiles), 2011
  - From the authors "existence of *practically exploitable vulnerabilities* that permit *arbitrary automotive control* without requiring *direct physical access*."
  - From physical, short-range, and long-range perspectives on components

Comprehensive Experimental Analyses of Automotive Attack Surfaces

Stephen Checkoway, Damon McCoy, Brian Kantor,
Danny Anderson, Hovav Shacham, and Stefan Savage
University of California, San Diego

Karl Koscher, Alexei Czeskis, Franziska Roesner, and Tadayoshi Kohno
University of Washington

Vulnerability Class	Channel	Implemented Capability	Visible to User	Scale	Full Control	Cost	Section
Direct physical	OBD-II port	Plug attack hardware directly into car OBD-II port	Yes	Small	Yes	Low	Prior work [14]
Indirect physical	CD	CD-based firmware update	Yes	Small	Yes	Medium	Section 4.2
	CD	Special song (WMA)	Yes*	Medium	Yes	Medium-High	Section 4.2
	PassThru	WiFi or wired control connection to advertised PassThru devices	No	Small	Yes	Low	Section 4.2
	PassThru	WiFi or wired shell injection	No	Viral	Yes	Low	Section 4.2
Short-range wireless	Bluetooth	Buffer overflow with paired Android phone and Trojan app	No	Large	Yes	Low-Medium	Section 4.3
	Bluetooth	Sniff MAC address, brute force PIN, buffer overflow	No	Small	Yes	Low-Medium	Section 4.3
Long-range wireless	Cellular	Call car, authentication exploit, buffer overflow (using laptop)	No	Large	Yes	Medium-High	Section 4.4
	Cellular	Call car, authentication exploit, buffer overflow (using iPod with exploit audio file, earphones, and a telephone)	No	Large	Yes	Medium-High	Section 4.4

Things Connected to Internet

- In summary, things suffer vulnerabilities when attached to the Internet
- A variety of causes, including
  - Flaws made accessible to adversaries when attached to Internet (*vulnerabilities*)
    - They were always there
  - Mismatch between programmer expectations and system deployment creates new vulnerabilities
    - The programmer did not provide defenses for this deployment
  - Trusted services may be compromised, which are new for the system
    - Thus, the deployment's trust model is invalid
- These problems are not unique to IoT, but may be exacerbated by the variety, dynamics, and uncertainty in IoT environments

Security Solutions

- Can the security community provide solutions to these fundamental cybersecurity problems?

Security Solutions

- Can the security community provide solutions to these fundamental cybersecurity problems?
  - Unfortunately, solutions are limited
  - Detect all flaws that lead to vulnerabilities (flaws accessible to adversaries)
    - Not fully automated
    - Some risks remain, so we aim to restrict what an adversary can control
  - Find all mismatches (between program and system deployment)
    - Programs and system distros developed independently
    - In some cases, methods can identify mismatches and add defenses to block exploitation
  - Minimize Trusted Computing Base (TCB) (use correct trust model)
    - Services need to support many mutually distrusting clients for many actions
    - In some systems, we can leverage alternative architectures that limit adversaries

Restrict Adversary Control

PENNSTATE 1855

- Suppose an adversary compromises critical, trusted software in IoT devices
  - Conventional kernels, microkernels, hypervisors, user-space servers, etc.
  - Linux, FreeBSD, MINIX, L4, Xen, BitVisor, file server, window server, ...
- Kernel rootkits are now becoming a serious threat to smartphone operating systems (e.g., Android)
  - CVE-2011-1823: an integer overflow bug in a daemon process on Android 3.0 enables an adversary to gain root privilege and install a kernel rootkit
- Can we restrict what exploits an adversary can perform?
  - 1 - Restrict code that an adversary can execute in supervisor mode
  - 2 - Restrict the way that approved code can be executed

IoT and Security

- Given the current situation in computer security, what should/ must the IoT community do?

IoT and Security

- Given the current situation in computer security, what should/ must the IoT community do?
  - Wait…

IoT and Security

- Given the current situation in computer security, what should/ must the IoT community do?
  - Wait…
  - Hope for the best…

IoTand Security

PENNSTATE 1855

- Given the current situation in computer security, what should/ must the IoT community do?
  - Wait…
  - Hope for the best…
  - Move ahead carefully
    - Identify attack surface (resources that may be accessible to adversaries)
    - Demand and use common solutions for defense
    - Address CPS-specific problems (physical inputs impact security)
  - Test with adversary in mind
    - Consider "best" action for adversary everytime your program accesses an adversary-controlled resource [STING, USENIX Security 2012]

Cyberphysical Systems

- Present additional challenges
  - May be more important
  - Need to understand/manipulate external environment
- Important
  - Smart house vs. my laptop
- External Environment
  - Physical to cyber
  - Cyber to physical

Physical to Cyber

- What if an adversary controls a sensor that measures the physical environment?
  - E.g., Temperature of my refrigerator

- Integrity
  - Can arbitrarily change the temperature measurement
- Traditional solution – Byzantine Fault Tolerance
  - Reasons about fault threshold – 2/3 of entities must be OK
  - Active attack may violate that limit
- How do keep 2/3 of sensors from compromise?

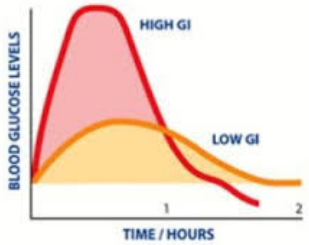
Physical to Cyber

- How do keep 2/3 of sensors from compromise?
- One approach: Make adversary devise more kinds of attacks
  - Animal defenses
  - Diversity and agility and deception

    - E.g., Make sensor look like one of several actuators to observer
    - E.g., Observer cannot tell which sensor is taking which measurements and change the mapping
- Goal: Detect compromised sensors before they can be leveraged with a high probability

Cyber to Physical

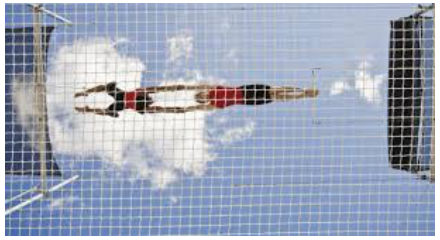
- What if an adversary controls an actuator that modifies the physical environment?
  - E.g., Insulin level



- Integrity
  - Can prevent changes from being implemented in physical
- Traditional solution – DoS Prevention
  - "Easy to detect and hard to prevent"
  - Potentially even harder to detect and prevent (Stuxnet)
- How to prevent compromise of physical from cyber?

Cyber to Physical

- How to prevent compromise of physical from cyber?
- Verifiably-safe behavior in face of DoS
  - Physical system should fail safe
  - Physical system needs internal recovery in lack of actuation
- Can detect/prevent maliciously-crafted commands
  - Commands leading to unsafe behavior require strong authentication
    - Two factor, challenge-response, etc.
  - Physical can verify command's safety locally



Privacy Issues

- In addition, how does CPS control access to private data?
- Proactive
  - Prevent leakage of medical information except to your medical professional (e.g., heart doctor for pacemaker)
    - What about in an emergency?

- Retroactive
  - Track leakage of medical information from medical professionals (e.g., ER doctors) to others
    - Possible to track closely enough? To punish?
- Good news – this is a general security problem

Defensive Options

- On which defense should CPS focus?
- Access control
  - Proactive defense (Allow/Deny)
  - Need to know whether to allow/deny in advance
- Auditing
  - Retroactive defense (Logging and log analysis)
  - Need to log/query the right stuff necessary to find violations after fact
- Agility
  - May be proactive or retroactive
  - But, may be costly or cheaper than above – e.g., Honey Passwords
  - Usually in addition to access controls/auditing

Other CPS Issues

- Heterogeneity
  - Impact of roles of sensing, reasoning, acting, reacting
  - Can we predict these roles and their attack surface (access to flaws) in advance?
- Dynamics
  - How map new elements to known/safe security concepts?
  - Can we infer mismatches in the changing environment?
- Uncertainty
  - How to address uncertainty in measurements?
  - Can we restrict adversary within some bounds?
- Goal is automation of the above

Security Analysis of Emerging Smart Home Applications

Earlence Fernandes, Jaeyeon Jung, Atul Prakash
Presented by: Gohar Irfan Chaudhry

IEEE Security and Privacy
24 May 2016

SmartThings System

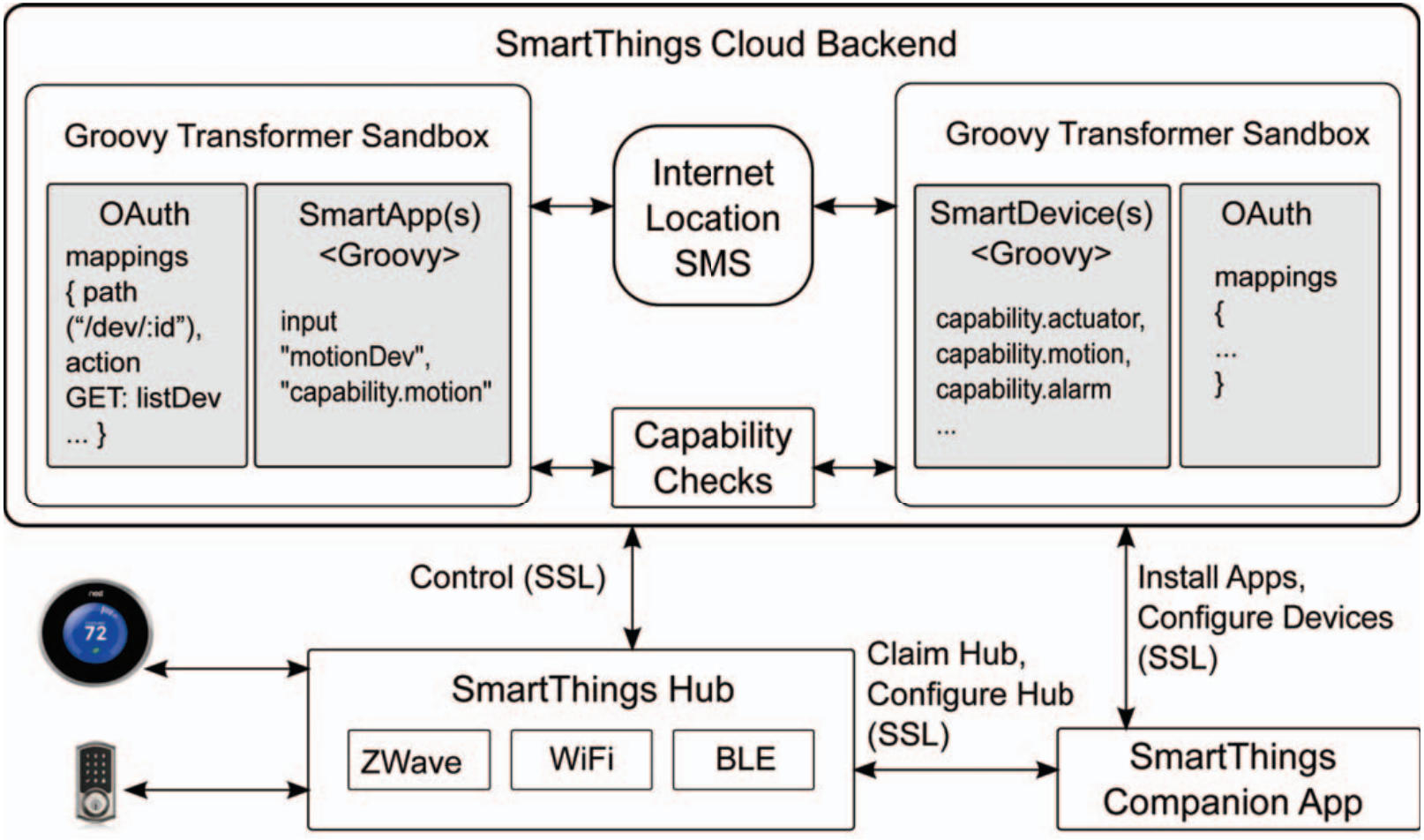


Fig. 1. SmartThings architecture overview.

SmartThings Vulnerabilities

PENNSTATE 1855

- Produced four PoC attacks
  - Injected malicious command to gain unauthorized access to a door lock
  - Snooped on setup of pin-codes for a Schlage smart lock, and leaked them using the unrestricted SmartThings-provided SMS API
  - Disabled an existing vacation mode SmartApp available on the app store using a spoofed event
  - Caused a fake fire alarm using a spoofed physical device event

One Attack

- Code injection attack
- Steps
  - Download app from App Store that requests user to authenticate to SmartThings and authorize a WebService SmartApp (written by same developer as 3rd party app) to access home devices
    - Obtain OAuth token from SmartThings deployment
    - Determine whether WebService uses unsafe Groovy dynamic method invocation
    - Inject command string over OAuth to exploit dynamic method invocation
- How/why is this attack possible?

One Attack Possible

- XSS-style attack to redirect OAuth token to the adversary
  - Seems like a flaw in OAuth
  - Link refers to authenticate SmartThings domain
  - But redirects to adversary-controlled URI
  - SmartThings automatically redirects the 6 character codeword
  - Adversary can then authenticate via Oauth
- A flaw in OAuth for SmartThings?

One Attack Possible

- Command injection phase of the attack
  - Does the app use dynamic method invocation?
    - Offline binary analysis - runtime
  - What command string should be injected?
    - Transmitted a payload to set a new lock code to the WebService SmartApp over Oauth
    - Allowed by capability for "SetCode" granted to the app
    - Commands are not sanitized in any way by the app
- What went wrong?

One Attack Possible

- Command injection phase of the attack
  - Does the app use dynamic method invocation?
    - Offline binary analysis - runtime
  - What command string should be injected?
    - Transmitted a payload to set a new lock code to the WebService SmartApp over Oauth
    - Allowed by capability for "SetCode" granted to the app
    - Commands are not sanitized in any way by the app
- What went wrong?

SmartThings Authorization

- Goals that the authors propose
  - Least privilege policy
  - Enforced over all security-sensitive operations
  - And others – input sanitization, authentication, etc.

SmartThings Authorization

- Goals that the authors propose
  - Least privilege policy
  - Enforced over all security-sensitive operations
  - And others – input sanitization, authentication, etc.
- Overprivilege
  - "capabilities" are associated with more than one operation
  - Capability.lock grants the ability to lock and unlock
    - Locking is relatively safe, so perhaps should not authorize both

SmartThings Authorization

- Goals that the authors propose
  - Least privilege policy
  - Enforced over all security-sensitive operations
  - And others – input sanitization, authentication, etc.
- In-Complete Mediation
  - APIs for some operations lack mediation entirely
    - Outbound Internet communications from SmartApps
    - SMS

Take Away

- New devices now connected to the Internet (and adversaries)
  - This has resulted in many problems in the past
  - But, stakes are high in CPS – impact on physical systems could be catastrophic
- To introduce new security challenges
  - Cyber-to-physical and dynamics/uncertainty of such computation
  - And still have to solve the traditional security problems effectively
- No trivial answers
  - Limit exploitation options, mismatches, and trusted computing base
  - New research in cyber-to-physical and physical-to-cyber
  - Hope it focuses on proactive and retroactive enforcement under dynamic and uncertain environments