SECURITY COMPASS WHITEPAPER

Internet of Things: Security Vulnerabilities and Challenges

Security Compass

It’s not quite Skynet, but the Internet of Things (IoT) is now pervasive in our lives.

In the military, data is collected from hundreds of sensors in the battlefield, from satellites, and radars, then processed for intelligence, while unmanned drones fight on our behalf. For consumers, virtual assistants like Amazon’s Alexa and Google Home can do our bidding to turn on lights in our homes or answer our doorbell from anywhere in the world. In the industrial world, Industrial Control Systems (ICS) designed to operate on isolated Operational Technology (OT) networks are now IP-based networks, presenting their attack surface to a broader audience, and internet connected tractors can determine if seed placement is accurate while guiding the vehicle around the field.

What do all these devices have in common? They are all connected to the internet and, like any other system, they all can include vulnerabilities waiting to be exploited by hackers.

Criminals Target IoT Devices

According to Symantec's 2019 Internet Security Threat Report, IoT devices experience an average of over 5,000 attacks each month, with routers and connected cameras accounting for over 90% of the attacks. Criminals recognize that IoT devices are soft targets, often designed for functionality with little regard for security, and poorly managed by many users. The consequences are millions of vulnerable devices providing simple attack vectors for adversaries.

We see examples in every area. The Mirai attacks of 2016 leveraged a design flaw in internet-enabled DVRs and IP cameras, allowing attackers to build a botnet army of millions of devices. They unleashed the botnet on DNS provider Dyn, resulting in the largest distributed denial of service attack ever seen, taking down Twitter, GitHub, Netflix, and other sites. While this was only an annoyance for those organizations and their customers, if the attack had instead targeted critical infrastructure or remote surgery, the results could have been fatal.

IoT devices experience an average of 5,200 attacks each month

Symantec 2019 Internet Security Threat Report

Vulnerable Devices Span All Industries

In the ICS world, a 2018 survey by Forrester found that "nearly six in 10 surveyed organizations using SCADA or ICS indicate that they experienced a breach in those systems in the past year". Roughly ten years earlier was the Stuxnet attack on an Iranian nuclear facility. This targeted scientists' laptops then moved laterally when it detected Siemens' SCADA systems on the network. The attack caused the facility's centrifuges to malfunction while overwriting log files to hide the attack, degrading or destroying over 1,000 nuclear centrifuges.

Advances in IoT for medicine have made lives better. Your doctor can now remotely adjust your treatment by communicating over the Internet with medical devices. Researchers have demonstrated vulnerabilities in their communications that would allow an attacker to modify dosages of medicine in wearable infusion pumps or reprogram a pacemaker.

Vulnerabilities in IoT devices even affects toys. Germany's regulatory office for telecommunications banned the children's doll My Friend Cayla calling it a "concealed surveillance device" after it was determined that a hacker could eavesdrop on conversations captured by the doll's microphone. The same was true for Mattel's Hello Barbie and Hasbro's Furby Connect.

"...nearly six in ten surveyed organizations using SCADA or ICS indicate that they experienced a breach in those systems in the past year"

Forrester Research

How Hackers Compromise IoT Devices

Like any other computing system, IoT devices are comprised of hardware and software and are designed to communicate with other devices or systems. Also like any other system, the designers and developers of IoT devices can make mistakes that result in exploitable vulnerabilities. This can be further exasperated in devices where maintaining low prices and accelerating time to market are critical, and security may be viewed as less critical.

Without the proper security practices to identify potential weaknesses and threats test controls for those weaknesses, the risk from these vulnerabilities can affect us all. This paper will look at the three primary attack vectors used by hackers: the device; the communications channels; and the cloud.

Attacking the Device

Competitive pressure drives many IoT device manufacturers to keep costs low. This starts with hardware/software stack. Arduino and Raspberry Pi are popular microcomputers– credit card-sized, single-board computers that run a variety of Linux distributions. These microcomputers can have both hardware and software vulnerabilities or may simply be poorly managed. In 2019, an unauthorized Raspberry Pi device connected to the IT network of the NASA Jet Propulsion Laboratory served as the entry point for hackers who stole almost 500 MB of Mars mission data .

Vulnerabilities that result from using out-of-date components (OWASP IoT Top 10 #5) is a preventable error during development, if one is diligent in checking those components against databases, such as The US National Institute of Standards and Technology (NIST) National Vulnerability Database (NVD). Since security is not a permanent state, one must also continuously monitor those sources for newly disclosed vulnerabilities affecting previously deployed devices.

Hardware Vulnerabilities

In 2018, high and critical privilege escalation vulnerabilities were disclosed in Raspberry Pi ARM-based hardware. These vulnerabilities provided a network/remote attack vector without requiring authentication. Also in 2018, a buffer overflow vulnerability was disclosed in C++ library for a variety of Arduino compatible boards. Raspbian, the board's operating system, has also had vulnerabilities reported against it.

Software Vulnerabilities

An open source operating system keeps costs low and is attractive to device manufacturers. That is reflected in The Eclipse Foundation's 2019 IoT Developer survey, showing 76% of IoT software developers basing their projects on various Linux distributions. While Linux is a good choice for any project, it too can include vulnerabilities and misconfiguration issues. In June 2019, Netflix's security research disclosed several TCP networking vulnerabilities in FreeBSD and Linux kernels. While a patch was issued with the disclosure, devices remain vulnerable until a user updates the software. Most organizations – and especially consumers – do this poorly.

Design Flaws

Other issues affecting the hardware and firmware on a device include hardcoded passwords or easily guessed default passwords. A recent study by ESET found that 15% of all routers use weak factory default passwords, making it simple for attackers to access and make changes to a network. The Mirai attack on Dyn exploited the fact that users were not required to change default passwords in the targeted devices. In addition, failing to encrypt credentials or sensitive data at rest can simplify an attacker's task.

Attack Communications

IoT devices typically require a communications channel to a network or cloud service. Your internet camera must access your local network and the Internet, and ICS sensors must relay information back to a controller, which in turn communicates with actuators. Insecure choices in design and configuration can make attacking the communication protocols simple. NIST's NVD lists over 60 vulnerabilities in the XMPP protocol alone. Other devices may rely on Bluetooth, but the 2017 BlueBorne attack leveraging Bluetooth vulnerabilities affected over 5.3 billion computers, phones, and other devices including Amazon Echo and Google Home.

8 zero-day vulnerabilities in Bluetooth protocol that impact more than 5.3 Billion devices—from Android, iOS, Windows and Linux to the Internet of things (IoT) devices—using the short-range wireless communication technology.

Man in the Middle Attacks

A man-in-the-middle (MITM) attack occurs when an attacker intercepts communications between devices or the device and its controller. This can allow an attacker to capture credentials and potentially modify data between devices. The consequences of MITM attacks can be catastrophic. We have seen this potential in the Jeep hacking demonstration by Charlie Miller and Chris Valasek, where they could override instructions from various sensors on a vehicle to cause the Jeep to slow down, turn the steering wheel, accelerate, or decelerate. In an ICS environment, one could imagine using MITM attacks to modify temperature data from a sensor to cause machinery or a server room to overheat.

Design Flaws

Attackers can also attempt to sniff and capture communications in communications protocols like ZigBee, Thread, Bluetooth, and others. If that traffic is not encrypted, it is trivial to capture credentials and other sensitive data. If it can be reverse engineered, attackers can modify and replay the traffic to take over the device. One doesn't even need to intercept communications with some attacks; simply spoofing a road sign can cause an autonomous vehicle to behave unsafely.

Relay Attacks

Rather than *replay* information captured, other attacks simply *relay* the data. An example of this has been seen with car thieves using key fob relays. Cars using proximity keys and keyless ignition only require the key to be near the vehicle to unlock doors and start the engine. In a key fob relay attack, one thief will use a transmitter to send a signal to the target car, prompting the car to reply with a request for authorization. This signal is sent to a second thief stationed close to the owner's valid key (outside a house, in an office, etc.). The valid key responds with its credentials, which are then relayed to the first thief and transmitted to the car, unlocking the car and allowing the vehicle to start.

Attack the Cloud

IT security used to be about defending the perimeter and keeping the bad guys out. Today, the application is the perimeter and presents a rich attack surface with each CPU core, device driver for Bluetooth, GPS, video, and USB port added. As IoT devices grow in number and complexity – Gartner estimates that over 20 billion IoT devices will be deployed by 2020 – this attack surface grows exponentially. In 2015, In-Q-Tel's CTO, Dan Geer, stated "If perimeter control is to remain the paradigm of cybersecurity, then the number of perimeters to defend in the Internet of Things is doubling every 17 months".

While many of the same secure coding practices used for other applications apply to IoT, OWASP has produced a Top 10 list specifically for IoT. This includes many of the issues covered in this paper, including weak passwords, insecure data transfer and storage, and use of insecure or outdated components.

"If perimeter control is to remain the paradigm of cybersecurity, then the number of perimeters to defend in the Internet of Things is doubling every 17 months".

Dan Geer
In-Q-Tel CTO

Regulatory Changes are Coming

Security of IoT doesn't just affect device owners. DDoS attacks, like Mirai, affect all of us. As IoT security has become a Board-level issue, it also has captured the attention of lawmakers and regulators. In 2018, California passed SB-327 and became the first state to regulate the security of IoT devices, requiring that manufacturers of any connected device sold in California to have "reasonable security features". At the national level, The IoT Cybersecurity Improvement Act of 2019 was introduced in the US Senate to require NIST to establish security standards for IoT device manufacturers covering secure development, identity management, configuration management, and patching.

There are, however, more granular guidelines for manufacturers. NIST first issued Special Publication 800-82, the Guide to Industrial Control Systems (ICS) Security in 2011, and updated it in 2015. The security architecture section of the standard provides guidance for developers on network segmentation, authentication and authorization, logging, incident response, and other items. Security controls standards cover a risk management framework and over a dozen individual controls.

Similarly, the International Electrotechnical Commission has published IEC 62443; security standards for Industrial Automation and Control Systems covering policies and procedures, system requirements, and component requirements in addition to detailed requirements for risk assessments and the software development lifecycle.

Device manufacturers should expect regulatory pressure to increase. The EU's General Data Protection Regulations (GDPR) covering personally identifiable information apply to any IoT devices managing this information. In the US, the Federal Trade Commission took action against D-Link under Section 5 of the FTC Act for "unfair or deceptive acts or practices", stating that the company misrepresented the security of its routers and Internet-connected cameras. As part of the settlement, D-Link must "establish and implement, and maintain, a comprehensive software security program" for 20 years. Among other activities, D-Link is required to follow reasonable security practices such as:

- "Engaging in security planning by enumerating in writing how functionality and features will affect the security of Covered Devices;
- Performing threat modeling to identify internal and external risks to the security of data transmitted using Covered Devices;
- Engaging in pre-release code review of every release of software for Covered Devices through the use of automated static analysis tools;
- Conducting pre-release vulnerability testing of every release of software for Covered Devices;"

How to Mitigate Risk in IoT

The first step in any security plan is to understand where risk exists. Threat modeling is an exercise that looks objectively at the design of the system to identify a list of potential threats, attack vectors, weaknesses, and risk mitigation strategies. Based on the results of the threat model, security requirements can be created and added to the project's functional requirements. This provides QA and security with a checklist to test against before deployment. A threat model will examine many factors, including:

- The criticality of the application and its security to your business goals
- The application architecture, hardware, and software stack
- The deployment environment (e.g., web facing, internal, IoT)
- The development language(s) (e.g., are buffer overflows possible)
- The software frameworks and other 3rd party components are used
- What type of data the application manages (e.g., sensitive data, personally identifiable information)
- What regulatory standards the application is subject to (e.g., PCI, HIPAA, GDPR)
- What hardware the application controls (Cameras, house locks, environmental)
- Application Layer Protocols Used (HTTP, XMPP, MQTT)
- Low-Power Protocols Used (ZigBee, Thread)

Automating Threat Modeling

Traditional threat modeling can be time-consuming, inconsistent, and incompatible with rapid development methodologies, like Agile and DevOps. Instead, many organizations are turning to automating the threat modeling process using SD Elements. SD Elements simplifies the threat modeling process by providing a consistent, thorough, and structured set of questions on the design and implementation of a system, then automatically generating security requirements, actionable tasks, sample code, and sample test plans to compliance with the requirements. Examples of requirements include:

- If the application is a client
  - Test to ensure that sensitive logs are not stored on the client
- If the application generates temporary files
  - Test to ensure that temporary files are cleared after the resource is used
- If the application has a user password
  - Require old passwords when users change passwords
  - Salt and hash stored passwords
  - Mask user passwords by default
- If using XMPP
  - Protect XMPP in-band registration
  - Check the integrity of MQTT messages
  - Limit the length and number of XMPP registration tags provided by IoT devices

Security Compass helps product management, development, and security work together to establish clear and appropriate requirements. By making these requirements visible early in the SDLC, organizations can more easily achieve regulatory compliance and lower development costs, avoiding unnecessary rework later in the development lifecycle.

SecurityCompass

Security Compass believes in a world where people can trust technology, and it is our mission to help customers proactively manage cybersecurity risk, without slowing down business. Our holistic program offerings are tailored to your organization's needs. Through advisory services, training, and SD Elements, our award-winning policy-to-execution platform, we set you up with all of the resources and tools your organization needs to develop secure software. Security Compass serves some of the world's largest enterprises, including 16 of the largest financial institutions in North America, South America, and Asia, as well as 4 of the largest tech companies in the world.

1.888.777.2211
info@securitycompass.com
www.securitycompass.com

@SECURITYCOMPASS
SECURITY COMPASS

OFFICES

GLOBAL HEADQUARTERS
1 Yonge Street
Suite 1801
Toronto, Ontario
Canada M5E 1W7

TORONTO
390 Queens Quay W
2nd Floor
Toronto, Ontario
Canada M5V 3A6

NEW JERSEY
621 Shrewsbury Avenue
Suite 215
Shrewsbury, New Jersey
USA 07702

CALIFORNIA
1001 Bayhill Drive
2nd Floor
San Bruno, California
USA 94066

INDIA
#4.07
4th Floor, Statesman House
Barakhamba Road, New Delhi
India 110001