An Architectural Risk Analysis for Internet of Things (IoT) Services

March 2019

As a leading technology and solutions development organization, the Alliance for Telecommunications Industry Solutions (ATIS) brings together the top global Information and Communications Technology (ICT) companies to advance the industry's most pressing business priorities. ATIS' nearly 200 member companies are currently working to address the All-IP transition, network functions virtualization, big data analytics, cloud services, device solutions, emergency services, machine-to-machine (M2M), cyber security, network evolution, quality of service, billing support, operations, and much more. These priorities follow a fast-track development lifecycle – from design and innovation through standards, specifications, requirements, business use cases, software toolkits, open source solutions, and interoperability testing.

ATIS is accredited by the American National Standards Institute (ANSI). The organization is the North American Organizational Partner for the 3rd Generation Partnership Project (3GPP), a founding Partner of the oneM2M global initiative, a member of and major U.S. contributor to the International Telecommunication Union (ITU), as well as a member of the Inter-American Telecommunication Commission (CITEL). For more information, visit www.atis.org.

Published by
Alliance for Telecommunications Industry Solutions
1200 G Street, NW, Suite 500
Washington, DC 20005

Printed in the United States of America.

Table of Contents

1 Executive Summary

The adoption of internet of things (IoT) services has accelerated enormously as multiple industries have recognized the extent to which IoT services can provide significant advantages to consumers, enterprises, and government institutions. Given the wide range of IoT-based services and products, their rapid entrenchment into daily life, and the increasingly sensitive and critical roles they can play, security must be a top consideration.

Protecting IoT services and applications requires a solid and well-founded set of security defenses that have been selected to provide the necessary (and—it is hoped—sufficient) safeguards against those threats deemed to pose the greatest risks to the service, its users, and the service providers. A risk assessment that identifies vulnerabilities in a quantifiable manner is a necessary first step in constructing a sensible IoT security posture. This paper has been written to address the fundamentals needed to perform risk assessments for IoT assets.

This report uses the *ATIS Security Architectural Risk Analysis (ARA)*[1] to establish a framework for assessing a generic IoT asset's cybersecurity risk. That asset might be an application, a service, or something else. Its primary function might be to collect and manipulate data, or it might be to perform some task, simple or complicated. It might be a standalone device, or it might work in coordinated fashion with a few or many other applications, services, or machines. It might be a low-level spoke in a wheel, or it might be a critical component of some vast mechanism. As an IoT "thing," it could be nearly anything, and for the purpose of this analysis, exactly what it is does not matter. It is simply "the asset."

This paper establishes the fundamentals that permit an examination of the risks to the asset from cyber-attacks that might be launched from anywhere, with a wide range of intents. The ATIS ARA methodology has been applied, with the following results:

1. This report provides a risk-assessment framework that is successful for:
   - Recognizing that IoT assets share many common features that differentiate them from non-IoT networked assets.
   - Identifying primary, secondary, and other IoT real-world assets, in accordance with the ARA methodology.
   - Establishing a complete set of consistent and useful attributes for each IoT asset. In working through the methodology, the authors have also established new criteria for determining the correctness of a set of attributes; namely, that they constitute a complete set—i.e., that for a given asset, the attributes listed are all there are; and that they be orthogonal in the sense that attacks on a given attribute must not bleed over to other attributes.
   - Providing a reasonable subset of the attacks to each of those assets. According to the ARA methodology, this is done by delineating the list of possible attacks against each of the asset's attributes based on how each attribute can be exploited (the so-called *exposures* of the attribute). This report shows how the ARA methodology is used to pinpoint the attacks and attack classes to which each attribute is exposed.

2. The report provides a simple though plausible example of how the ARA can be applied to an IoT asset.

One major benefit of an ARA risk assessment is that its results identify specific security mitigations that can address the most serious threats. As a result, it can suggest the best expenditure of time, money, and resources to fortify an asset to mitigate the most serious risks. In the case of IoT assets, the size, complexity, and perhaps cost to the user all put strict limits on how much development time, effort, and money can be directed towards security. Thus, the capability to assess the risk provides immense value to designers, architects, and planners as they go through their respective tasks of bringing an IoT asset into existence. The framework developed in this report offers a mechanism for using the ARA methodology to make that happen more directly because it has been tailored to the IoT landscape.

[1] *Cybersecurity Architectural Risk Analysis Process*. ATIS-I-0000057. May 2017. Alliance for Telecommunications Industry Solutions; Washington, DC.

2 Introduction

The emergence of IoT-based services is already creating explosive demand for new devices and applications. According to *Forbes*,[2] a 2017 survey predicted 20 billion connected devices globally by the end of 2018, with steady (rather than explosive) growth through the year. *Business Insider*[3] projects that "there will be more than 55 billion IoT devices by 2025, up from about 9 billion in 2017." It goes on to predict "that there will be nearly $15 trillion in aggregate IoT investment between 2017 and 2025, with survey data showing that companies' plans to invest in IoT solutions are accelerating."

The growth forecasts in terms of dollars are probably a more telling indication of the explosion of interest in and adoption of IoT than raw numbers of devices would ever be. The devices themselves, which include connected cars, machines, meters, wearables, and other consumer electronics, will be connected to the internet or to private network environments and will need robust security mechanisms (such as the capability to clearly assert an identity that can be authenticated by the network and/or application appropriate for that device).

In some cases, the network operator's role in delivering IoT services is simply to provide connectivity and there is no direct technical or business partnering between the operator and the IoT service provider. In other cases, the network operator may take a more active role where the IoT service includes technical and business aspects under the control of the network operator.

IoT device types range from simple, small sensors to complex, large systems such as connected vehicles, which contain dozens of IoT devices. Regardless of the form factor or level of sophistication, devices must connect to IoT application servers or client devices via one or more intermediate networks. For example, the device may connect via private or public networks using a variety of connection mechanisms such as:

- Wired access via a physical connection (e.g., Ethernet cable to a local network).
- Wireless unlicensed private access (e.g., Wi-Fi or Bluetooth to a private network).
- A public licensed spectrum wireless access using standard 2G, 3G, 4G, or 5G radio technologies.
- Public wireless access provided by network operators using unlicensed spectrum technologies such as Wi-Fi, Citizens Broadband Radio Service (CBRS), and MulteFire™-based networks.
- Other wide area networks using various Low Power Wide Area Network (LPWAN) technologies.

In some cases, the device may connect to the network via an intermediate gateway device.

Given the wide variety of device types and access methods coupled with the large numbers of devices being deployed, network security concerns voiced by many experts[4,5,6] are well founded. This paper looks more closely at network-related security aspects of the IoT domain, with a specific focus on the security risks associated with IoT devices and services.

---

[2] Newman, Daniel. "The Top 8 IoT Trends for 2018." *Forbes*. December 19, 2017.

[3] Newman, Peter. "There Will Be More Than 55 billion IoT Devices by 2025—These Are the Biggest Drivers For Adoption." *Business Insider*. July 27, 2018. https://www.businessinsider.com/internet-of-things-report?op=1

[4] Meola, Andrew. "How the Internet of Things Will Affect Security & Privacy. *Business Insider*. December 19, 2016. https://www.businessinsider.com/internet-of-things-security-privacy-2016-8

[5] Palmer, Danny. "IoT Security: Where Do We Go from Here?" *ZDNet*. November 13, 2018. https://www.zdnet.com/article/iot-security-why-everyone-needs-to-step-to-ensure-the-security-of-the-internet-of-things/

[6] Zorz, Zeljka. "IoT Security: The Work on Raising the Bar Continues." *HelpNetSecurity.* August 22, 2018. https://www.helpnetsecurity.com/2018/08/22/iot-security-challenges/

2.1 Document Purpose

With an awareness of IoT security on the rise, methods to provide protection to the devices, the services, the data, and the users of all three must start with an understanding of the risks that IoT-based technologies introduce or exacerbate. It is generally agreed that a risk assessment is a necessary step that should predate the design of solutions and the creation of solutions architectures, so that correct security solutions can be designed in up front and a comprehensive security architecture can be crafted to dovetail with the overall solution architecture itself. This document provides a starting point for assessing the risks associated with IoT solutions. To do so, it applies the ARA to general IoT solutions involving network operators. Through this process, threat modeling techniques are applied to ensure proper security considerations are part of the solution by design.

No part of this document should be taken as normative. Its purpose is to document practices that may be helpful to the development of good solution security. As each situation is different, it is necessary for the security approach to be chosen by the parties involved appropriately for their service, priorities, and circumstances.

2.2 Scope and Context

The ARA methodology is expected to help network operators, application providers, and their third-party partners and suppliers to assess the robustness of their IoT architecture by identifying key points where security controls are needed to thwart potential threats and the associated risks. Prior to applying the ARA process, it is necessary to establish a context that supports the application of the process.

IoT solutions involving network operators may engage four players or actors, illustrated in figure 2.1, in providing the IoT service:

- The IoT device management entity handles processes such as providing device configuration and fault-management services. The IoT device might be a single physical device, a gateway, or a controller for a multitude of sensors or actuators. The IoT device might be directly connected to the network operator via a WAN interface to cellular (2G, 3G, 4G, or 5G). Or it might be connected via a LAN, which in turn connects to a wireless or wireline network via a gateway/modem element.
- The network operator that provides the network service.
- One or more identity providers that certify the identity of the device for network, application, or management authentication and/or authorization purposes. For WAN access, this entity is often the network operator. However, roaming cases exist where a home network provides authentication services to a visited network. Separate identity providers may also exist to provide application- or management-specific services.
- The application provider that provides a service based on IoT device communication. The application provider will often utilize servers to provide the IoT application. Client devices may also be enabled on behalf of the application domain to connect directly to IoT devices.

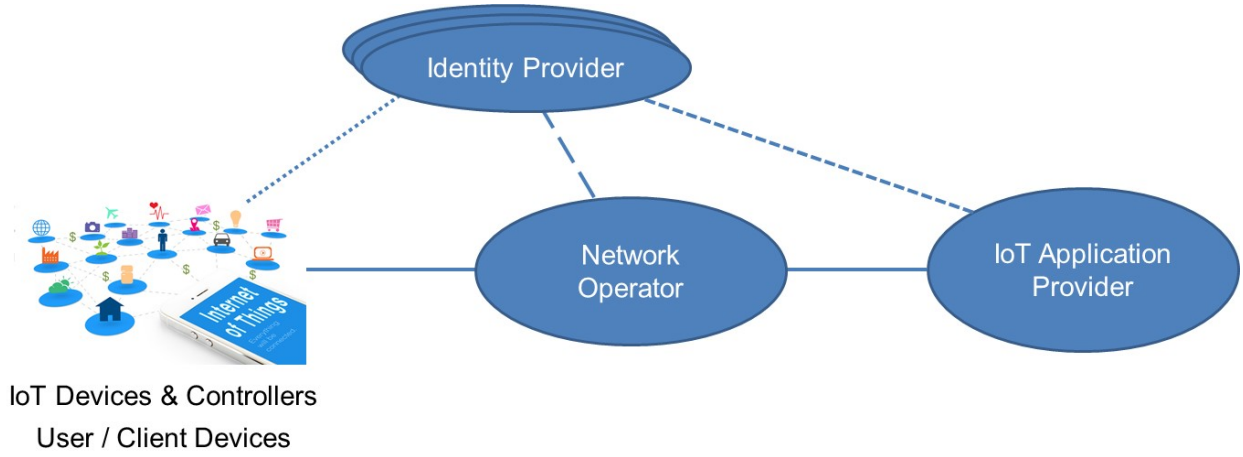


Figure 2.1 – IoT Key Actors

The focus of this document is on the devices and network aspects used to provide IoT services and applications. As such, from a security perspective, we use the terms “IoT service” and “IoT application” interchangeably. In most cases, we use IoT service to refer to services and capabilities provided by a network operator and IoT application to refer to the role of IoT-specific servers.

The ARA process will leverage the perspective of each of the four actors in support of an IoT service or application. The focus of this analysis will be on the technical aspects of secure communications between the IoT device and the IoT application provider (and associated servers and client devices utilizing the IoT application).

We define an IoT service as a service that enables devices in the access network that are not directly controlled by a human to communicate with people- or server-based applications across a WAN. An IoT device is a device or subsystem of a device that is used to monitor and/or control some aspect of a physical object. As such, we say an IoT device can provide sensed information (i.e., it can monitor/measure), enable the control of some actuator (i.e., control the device), or both.

IoT services share many common elements and infrastructure with other classes of services such as interactive voice/video-based services and web services. As such, many of the security aspects noted in this analysis will be common to these service classes. However, IoT services share some unique aspects that present distinct security challenges, as well as opportunities. Security challenges include:

- IoT devices measure/monitor and/or control aspects of physical objects. As a result, IoT security vulnerabilities provide opportunities for attackers to interfere with the physical safety of people and things that might depend on these objects. For example, a compromised heating and air conditioning IoT device can cause physical discomfort to people in the facility. It also can cause physical damage by, say, turning the heat off in freezing temperatures, resulting in burst pipes and/or system failures. Compromise of control systems is of concern for IoT devices used in critical infrastructure.

- Many IoT devices may utilize network communication modes (such as P2P, along with network address translation (NAT) traversal technologies). These may present new opportunities for attack and new classes of vulnerabilities.

- IoT devices and applications may choose to utilize authentication and authorization delegation models to enable more flexibility in exchanging IoT data. These models offer new challenges because, in most cases, delegation protocols were not specifically designed for IoT architectures. As such, P2P mechanisms are often used to exchange data or control information between a community of IoT components. However, P2P authentication can create challenges because, unlike a normal client-server connection—where one security relationship exists between the client

and the server—a P2P configuration requires establishing pairwise security relationships between all devices and servers within the community of interest. This creates additional challenges in managing the distribution of robust credentials in a scalable and secure manner within the P2P community.

Opportunities include the following:

- IoT devices are more likely to be purpose-built for their intended function. As such, they are more likely to have known software images and configuration structures. This makes it easier to verify both software and configuration remotely using secure attestation solutions.

- IoT services may leverage unique network capabilities (e.g., a 4G/5G IoT network slice). As such, they can take advantage of the natural segregation attributes of the network capability to enhance security.

In providing this IoT ARA, we create a framework to enable consideration of many of these IoT unique attributes.

In this document, we often refer to the set of devices and network elements that provide an IoT service as an IoT system. For example, an IoT system may be composed of one or more IoT devices and IoT controllers connected via one or more LAN segments and a WAN to servers and client devices.

2.3 Target Audience

The process is intended for those involved in planning, designing, engineering, evaluating, and implementing information and communications technology (ICT) solutions. Backgrounds in general computing, networking, and security will help the reader to apply the process effectively and gain the best results.

3 IoT Security Landscape

There are several challenges to securing IoT. Some IoT providers do not have security in their budgets. Some devices do not have the memory or computing power to support the security they really need. Some designers do not recognize the potential for abuse and do not design for security. The list goes on. Add in the prevailing paradigm of building software update capabilities into almost every modern device, and you end up with billions of devices that have inherent security weaknesses.

What, then, does one do when that target itself can be, at best, only imperfectly defended? Several well-used options present themselves:

1. **Fortify the access routes into the device to reduce or eliminate traffic having malicious content.** For example, IoT traffic could be subject to network-based monitoring and profiling functions that could detect anomalies and filter traffic. Fortifying the access routes, though, could be thought of as akin to fortifying the physical routes to a bank by adding checkpoints, inspections, and significant challenges that the robber must overcome to arrive at the target. This has the unwanted effect of seriously inconveniencing honest citizens having legitimate business at the bank, and, except possibly in the case of Fort Knox, is impractical. Similarly, trying to construct electronic roadblocks within the communications infrastructure across which IoT traffic passes could create unacceptable roadblocks for ordinary or legitimate traffic while providing only moderate protection to the asset—and that does not even take into consideration the time, effort, and expense of fortifying existing communications protocols, defining new security requirements for the next line of protocols, managing the security of all the working pieces, and making it all work on a per-customer basis. Nevertheless, in situations where the IoT traffic can be segmented, network segmentation mechanisms such as those provided by VPNs can provide a level of “fortification” when the device needs to communicate only with other devices within the defined VPN.

2. **Increase the difficulty of getting into the device itself to reduce the number of successful intrusions.** This is akin to making the bank accessible to only those people having a legitimate and demonstrable reason to enter it. In principle, this should be an effective defense. But in the real world, the default password and other nominal security choices reduce it to a trifle akin to letting only those people into the bank who know today's secret word. This is the same as yesterday's secret word, for as far back in time as the oldest customer can remember. To make the current authentication and authorization strategies work is a huge undertaking that has enormous problems with scaling. Managing a few hundred passwords and access permissions is challenging enough, but it's trivial compared to managing a few billion.

3. **Create layered defenses around the device that produce independent yet redundant safeguards.** Defense in depth is a good strategy that can be effective when the protected item is worth the expense, time, and effort it takes to make the strategy effective. Fort Knox is worth all three, but the average bank simply is not Fort Knox, so it's impractical to borrow the Fort Knox strategy. A scaled-down version that the bank can afford is probably not going to do the job because the protective measures are independent and meant to bolster each other. It is easy to imagine the problems involved in scaling down multiple pieces of an interlocking mechanism and still having it work. Also, most IoT devices are not Fort Knox and thus can't justify the expense of defense in depth. Those devices, such as the IoT components for connected cars and others on which human life could depend, might be good candidates for defense in depth. However, it is hard to imagine that anything but a small percentage of the 30 billion-plus expected IoT devices will fall into that category. One must wonder whether defense in depth is a good general strategy for the breadth of IoT security needs.

4. **Decrease the attacker's perception of the device's value and capabilities.** At a conceptual level, a really easy way to do this is to remove the capability that so many hackers and cyberwarriors exploit: Do not allow the device to be remotely reprogrammable. However, this runs directly afoul of the current paradigm that permits remote reprogramming of devices, so it's likely to be dismissed by vendors or users. A possible fallback defense is to make the job of accessing and reprogramming IoT devices so difficult that subverting it is not worth the attacker's time and effort. However, considering that even the simplest devices can made into a botnet, one wonders what steps could be taken to prevent such takeovers within a manageable budget.

The discussion so far suggests that these options require solving some truly difficult problems in scaling; in coordination among devices, vendors, and users; and in security management at the operations level.

It is important to include an assessment of the "whys" of IoT attacks so that we might better understand the nature of the threats by knowing what an attacker can reasonably hope to attain. This assessment should address both simple and complex IoT devices so that differences in attackers' motives might be evaluated based on what the device should be capable of.

Some reasons for attacking IoT devices are to use them to:

- Launch a hyperscale DDoS attack.
- Act as a command-and-control point for externally sourced attacks.
- Provide a platform for lateral movement inside an otherwise secured network.
- Provide unauthorized services to malicious user community, such as crypto mining or hosting illegal content.
- Destroy services, such as by bricking the router or other device.
- Acquire unauthorized control of another device, such as to slam on the brakes on a vehicle.
- Exfiltrate data.
- Manipulate sensor data to hide real conditions, such as masking the fact that a centrifuge is spinning at dangerously high speeds.
- Acquire unauthorized data, such as taking control of a smart speaker's mic to eavesdrop.

Ultimately, even the wide range of security strategies and intercompany relationships that may be deployed to secure an IoT system cannot provide absolute protection from all attacks over the life of its service. It is not so much a question of how to secure the IoT system from all possible attacks. Instead, it's a question of which security capabilities should be employed to best secure the system at a risk level that is appropriate for the value of the service or device. In that light, it seems clear that a risk analysis process is necessary in developing a security strategy. The ARA methodology is a workable, scalable process that we have selected to model IoT risk.

4 ARA Process Elements & Overview

The ARA methodology involves defining the attack surface of solution assets, assessing the risk to each asset and (optionally) assessing how well the associated threats are mitigated through security controls. The high-level steps are described in Figure 4.1 which divides the process into three broad activities: architecture discovery, threat identification, and risk analysis.

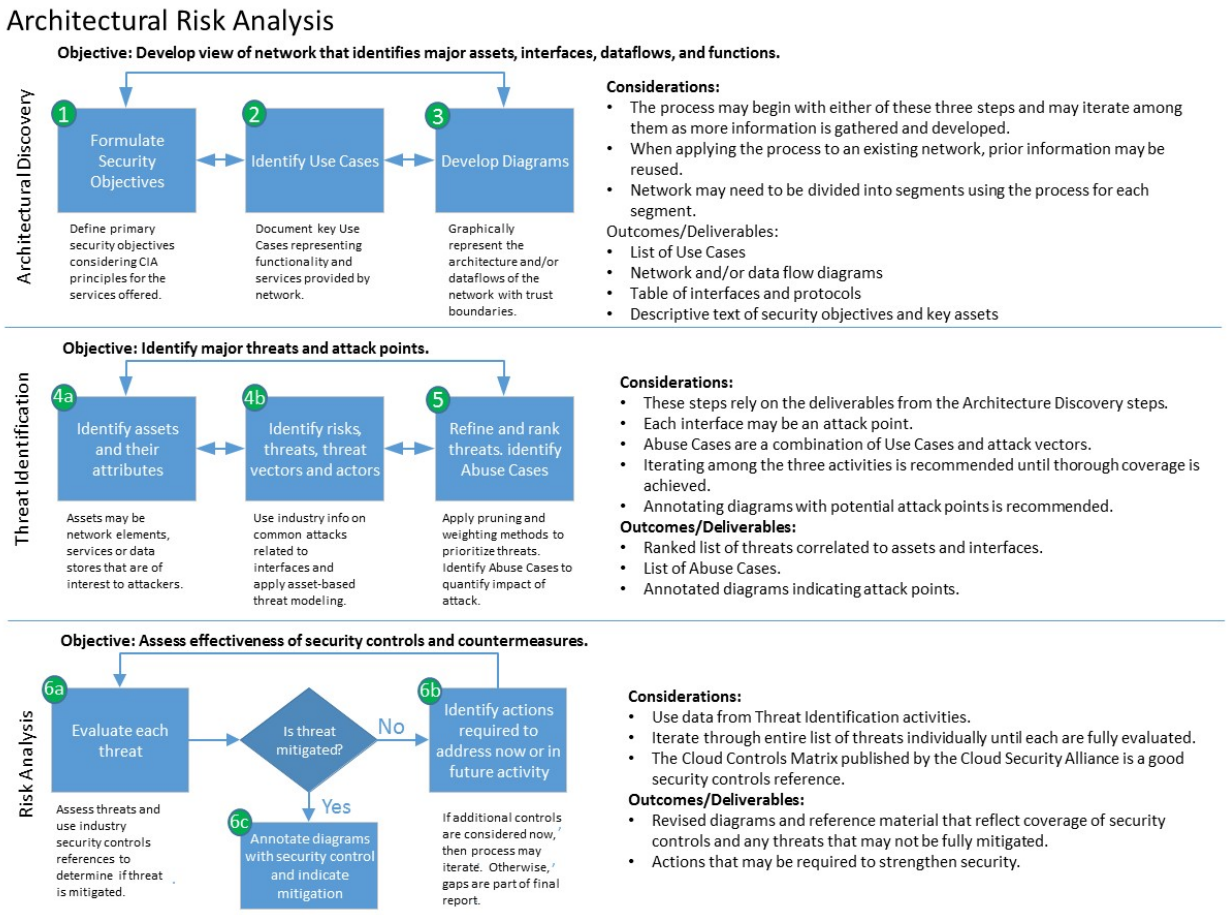


Figure 4.1: Process Steps

The process may begin at any of the three steps within the architectural discovery. The primary objective of architectural discovery is to develop a view of the network that identifies major assets, interfaces, dataflows, and functions. When applying the process to an existing network/system, prior information may be reused. It is often useful to iterate among these three steps as more information is gathered and developed.

For large, complex systems, architectural discovery can be applied in a hierarchical manner. That is, the network can be divided into segments with each segment being analyzed separately. The segments can then be brought together for a complete analysis.

The deliverables for the architectural discovery activity include a list of use cases, network and/or data flow diagrams, tables of interfaces and protocols, and descriptions of security objectives and key assets.

The primary objective of the threat identification activity is to identify the major threats and attack points relevant to architecture. As such, it relies on the deliverables from the architectural discovery work. Annotating diagrams with potential attack points is recommended.

Abuse cases are a combination of the use cases and the identified attack vectors. When doing this analysis, it is important to note that each interface may be an attack point. Iterating between abuse case and threat model efforts is recommended until thorough coverage is achieved.

The deliverables for the threat identification activity include a ranked list of threats correlated to assets and interfaces, a list of abuse cases, and annotated diagrams indicating attack points.

The primary objective of the risk analysis activity is to assess the effectiveness of security controls and countermeasures that may be used. This activity utilizes the outputs from the previous activity to iterate through the entire list of threats individually until each has been fully evaluated. The deliverables for this activity include a set of revised diagrams and reference material that reflect coverage of the security controls and any threats that may not be fully mitigated along with the set of actions that may be required to strengthen security.

Table 4.1 provides an expanded description of each step, identifies key contributors, and defines entry and exit criteria for each step.

Table 4.1 – Process Steps

Step	Short Description	Long Description	Contributors	Input	Output
1	Security Objectives	Document security objectives that correlate to the functionality and services provided by the network. Correlate to Confidentiality, Integrity, and Availability (CIA). Define third-party Trust Models.	Lead designers or architect, including those representing third-party components.	Project plan.	Documented security objectives.
2	Use Cases	Identify/document functional Use cases with security impacts Work includes relevant third-party use cases. Identity assets and their attributes that may be of interest to attackers.	Product Management, lead designers, architect, or engineers.	Project goals, requirements, or pre-existing use cases.	Use cases and assets that may have security impact.
3	Network Diagrams	Develop (or locate) an architectural diagram that depicts the major components, interfaces, and trust boundaries of the solution. Ensure that data flows, trust boundaries, and protocols of critical data paths are identified. Work includes	Lead designers or architect.	Existing documentation or high-level scenarios/use-cases of functional operation.	Architectural diagram depicting the solution.

Step	Short Description	Long Description	Contributors	Input	Output
		relevant third-party components.			
4	Assets, Risks and Threats	Identify assets and attributes along with the associated risks, threats, threat vectors, and actors. Document risks and threats that may target assets and associated interfaces and protocols.	Developer or engineer.	Threat library derived from industry research.	Annotated architectural diagrams and use case documentation with threats identified.
5	Abuse Cases	Document how use cases could be abused, identifying the overall solution impact and if possible, the methods of misuse.	Lead designers or architect.	Design and architecture documentation.	Ranked list of threats and abuse cases and list of key assets and interfaces
6	Risk Analysis and Threat Mitigation Plan	Does each threat have an associated security control to mitigate it? Determine additional controls and/or research needed to mitigate threats including those which are manifested via third-party components	Developer or engineer.	Annotated architectural diagrams and use case documentation with threats identified.	Description of how each threat is/is-not mitigated. Annotate diagrams with security control and indicate mitigation. Actions to strengthen security.

4.1 Architectural Discovery

The objective of the architectural discovery activity is to develop a view of the solution under analysis that identifies major assets, interfaces, data flows, and functions. These items comprise the core components that contribute to the network's attack surface and are critical to the remaining steps in the process. Components provided by third parties that can contribute to the attack surface of the solution should also be explicitly identified along with expectations for trusted and un-trusted assets, interfaces, data flows, and functions for such components.

4.1.1 Security Objectives

Clear security objectives are required to provide an operational context for developing a threat model and conducting the detailed analysis. Security objectives, or goals, should be as specific as possible so that analysis can clearly assess whether the goals have been met. They also must reflect the perspectives of the involved actors discussed in section 2.2. Table 4.2 summarizes the perspectives and goals.

Table 4.2 – Perspective and Security Goals

Perspective	Security Goals
Device Management Entity	• Ensure the device is operating under the correct software load and configuration data and that this data has not been compromised. • Ensure the device behaves as specified when configured properly. • Authenticate the network(s) and application provider(s) to ensure the device is communicating to known trusted entities. • Ensure that access to IoT application resources and controls are properly authorized. • Ensure the confidentiality, integrity, and availability of communications with network and application entities.
Network Operator	• Fortify the integrity of the network against devices not behaving as expected. • Ensure the confidentiality, integrity, and availability of network communications between the application provider and the device. • Prevent “theft of service” – i.e., ensure proper charging for a network service. • Avoid being party to botnet/DDoS attacks initiated by devices connect to the network. • Ensure traffic segmentation for VPN services.
Identity Provider	• Ensure uniqueness, correctness, and integrity of device and application credentials. • Ensure secure device and application authentication and authorization to resources according to defined policies.
IoT Application Provider	• Authenticate the IoT device as a known device to the application provider – both servers and application client devices. • Ensure that application resources are available only to authorized parties based on policies. • Ensure the confidentiality, integrity, and availability of communications between the IoT device and application server/client devices. • Ensure the confidentiality, integrity, and availability of communications between the network operator and the device (e.g., for mobility devices, the network operator may provide application programming interfaces (APIs) to allow the application provider out-of-band device access for functions such as paging).

In addition to the more specific objectives listed in the table, broader objectives can also be considered to help balance the risk/cost equation when considering threat mitigations. For example, it is reasonable to consider:

- Objectives to address the risks to corporate brand, reputation, customers, and network.
- Objectives to help manage the security relationships and responsibilities with any partners associated with the IoT solution. Specifically, IoT solutions that utilized identity and/or authorization delegation mechanisms might engage partners that may act as identity providers, and provide supporting IoT device data/controls or application functions. Establishing such objectives helps answer questions such as:
    - How much trust can one have in the IoT partner's level of security?
    - Will the IoT partner share the burden of security?
    - Does one wish to minimize security dependencies on the partner?
- Objectives relative to human resources associated with the specific IoT solution. An example is the level of security training required for each job category for people supporting the IoT service.
- Objectives to ensure that the software and hardware supply chains used to instantiate the product are secure.

For the purposes of this analysis, objectives that directly impact the technical aspects of secure communication between the IoT device and application server or client device are of greatest interest.

4.1.2 Use Cases

Once security objectives have been documented, functional use cases and their associated assets and attributes that may be of interest to attackers can be identified. The use cases later in this section illustrate the need for security within an end-to-end networked IoT solution from the IoT device to the IoT application server, along with the need for secured interfaces. The use cases are summarized for brevity, highlighting important security functions that are relevant to the described IoT solution.

As with most end-to-end network solutions, solution functionality can be divided into two major classes:

- **Management functions**, which are the standard fault, configuration, accounting, performance, and security (FCAPS) functions typically needed to properly manage components within the system.
- **Operational functions**, which include those capabilities needed to provide the intended function of the IoT system.

Typically, the IoT device will be used to monitor, via sensors, the state of the target object, along with the optional capability to control some aspects of the object. Both monitor and control functionality will likely have requirements specifying the performance, confidentiality, integrity, and availability of the functions.

Additionally, each class will require:

- Basic network access functions to authenticate the device with the appropriate management or operational entity.
- Network transport services to carry data objects between the device and the appropriate management or operational entity.
- Optional, out-of-band control functionality to allow the application or device to get contextual information about the IoT system or to control IoT system behaviors for the specific device/server instance. Examples include network location services, active/idle status, paging services, device activation, or custom data routing (e.g., to a portal).

IoT devices may use a wide variety of authentication mechanisms when attaching to a network. These mechanisms often depend on the access type or technology and can have varying security constraints and requirements. In addition, many simple IoT devices utilize a gateway element to consolidate, monitor, and control functions of many IoT devices in a residence or enterprise setting. In such cases, the gateway itself is the IoT device seeking network access. In the following access use cases, we categorize based on access type.

Given the wide variety of access types, we further partition the access space into LAN- and WAN-based connectivity. Typically, LAN access will utilize direct-wired connectivity or short-range radio technologies. The short-range radio segment uses unlicensed spectrum with a typical range of up to around 100 meters, such as Wi-Fi, Bluetooth, and ZigBee. The WAN access category consists of devices using cellular connections (2G, 3G, 4G, and 5G), as well as unlicensed low-power technologies, such as Sigfox, LoRa, and others.

The use cases in this section are structured into four domains: IoT device, LAN, WAN, and application-related functions. For each domain, specific functions are identified. The use cases are created by observing how each function is used in the context of IoT. Later sections will identify how the function can be attacked, the attributes and assets, and suggest possible options to mitigate the attack.

4.1.2.1 IoT Device Functions and Use Cases

FUNCTION/ATTRIBUTE	USE CASE – HOW THE FUNCTION IS USED
ACCESS	
Access (L1) Physical	• Connected IoT devices will have a physical Layer 1 communication link to enable connectivity. Low-end devices may not support security at upper layers of the protocol stack and, as such, may require analysis of physical layer attributes to enhance security (e.g., channel estimates). • Devices may also have physical switches or controls to alter basic configuration information on the device. Physical access to the device will also occur during supply chain activities.
Access (L2) Link	• Connected devices will have a L2 link layer access protocol interface using wired or wireless. Authentication may or may not be required on L2 depending on protocol choice and configuration (see LAN functions and attributes below). • Low-end devices may not support security at upper layers of the protocol stack and, as such, may require analysis of L2 link attributes to enhance security (e.g., packet size, timing, use of optional fields).
Access (L3+) Network	• WAN connected devices will have a L3 network layer access protocol interface to provide access to IoT applications. L3 access may include restricted local access to allow on-site configuration and programing of the device.
Authentication and Authorization (all layers)	• Connected devices should have capabilities to authenticate and authorize resource access at both access and application layers as needed

	and for management and operations functions as needed.
TLS Support (or similar)	• Connected devices may support Transport Layer Security (TLS) for secure network communication providing some level of privacy and data integrity.
Delegation Support for Authentication and/or Authorization	• Connected devices may participate in delegation models to allow another device or server to access data or assert an identity via a separate identity relationship with a common identity provider.
MANAGEMENT FUNCTIONS	
Programmable	• The software image of a device may be alterable/updatable. This allows for the introduction of new features or fixes to existing code. Programmability as a built-in function is generally deemed essential from a security perspective to ensure that discovered vulnerabilities can be corrected in the field. A device may be programmed remotely over a network connection or may be done locally.
Configurable	• Configuration refers to data provisioned on the device to enable the device to operate in a specific environment. A device may be configured locally or remotely over a network connection or may be done manually.
State Retrieval	• A network management system may be able to retrieve device state information (i.e., information acquired by the device associated with its state of operation).
OPERATIONS FUNCTIONS	
Service – Object Control	• An IoT device may function as a control point to provide specific control functions to an object. For example, an IoT thermostat might allow a remote party to change the temperature.
Data – Monitor Object	• An IoT device may function as a data collection device, collecting and reporting on metrics associated with its local environment as per the design of the device.
INTERNAL (COMPUTE)	• Connected IoT devices can have computing capability that includes:

	• CPU – quantity and speed • Memory – quantity and type • Storage – quantity and type • The IoT device will also have a service life determined in part by the on-board CPU, memory and storage functions.

Table 4.1.2.1 – IoT Device Functions and Use Cases

4.1.2.2 LAN Functions and Use Cases

FUNCTION/ATTRIBUTE	USE CASE – HOW THE FUNCTION IS USED
WAN GATEWAY	• A WAN gateway function provides WAN connectivity to IoT devices on a LAN. This function may include a modem to provide L1 connectivity to the WAN. L2 and L3 functionality are generally provided by a router associated with the WAN gateway function, with associated services such as firewall (FW)/network address translation (NAT), access control lists, and other routing services. This function usually supports a wide variety of LAN access technologies, as well as Universal Plug and Play (UPnP) capabilities.
IoT CONTROLLER	• IoT controllers provide interface and aggregation functions for a set of IoT devices on the LAN. Many IoT devices, such as light bulbs, may only support very basic access mechanisms and thus require a controller to fully operate and maintain the device. In addition, coordinated multi-device control, as well as local control capabilities, are often implemented via a local IoT controller device in the LAN.
MANAGEMENT FUNCTIONS	
	• Both the WAN gateway and IoT controller elements may have management functions accessible (often separately) by a network entity (either the network service provider, the IoT application provider, or delegated IoT user), as well as locally by an IoT user. In general, the three primary management functions noted for IoT devices are also applicable to these LAN components: o Programmability o Configurability o State retrieval

	• For management access to these functions, the L3 access technology functions listed in the IoT device table are applicable. Specifically, the management interface on these elements: o Should support authentication and authorization mechanisms for access and management of specific resources. o May support TLS for data privacy and integrity (or other similar technologies). o May support delegation models for authentication and/or authorization to selected resources.
OPERATIONAL FUNCTIONS	
WAN Gateway	• Transport services – The WAN gateway's primary function is to provide a transport interface between the LAN and the WAN to enable broader network connectivity. • NAT/FW and associated FW capabilities, which may include IPv4/IPv6 functions. • UPnP support – Commonly available to enable devices on the LAN to open ports to the WAN. • VPN support – VPN termination point for network VPNs including IPsec and TLS-based VPNs. • Other routing services – Enable specific routing rules for selected services/ports and/or devices.
IoT Controller	• IoT controllers include all operational capabilities of an IoT device. • May aggregate operational control of a set of IoT devices connected to the IoT controller via a wide variety of LAN access technologies. • May include functions to enable coordinated control of IoT devices based on configured sequences in time.
LAN ACCESS TYPE	
	• Wired • Wi-Fi • Zigbee • Bluetooth • Other
LAN CAPABILITIES	

	• Discovery (e.g., Simple Service Discovery Protocol [SSDP]) • Network segmentation (e.g., VLAN Support) • Quality of service (QoS) (e.g., 802.1q/PCP)

Table 4.1.2.2 – LAN Functions and Use Cases

4.1.2.3 WAN Functions and Use Cases

FUNCTION/ATTRIBUTE	USE CASE – HOW THE FUNCTION IS USED
Network Transport	• The WAN's primary function is to provide transport connectivity services at L3 to devices and servers on the WAN. • Network NAT/FW devices commonly deployed between the access edge and the WAN core network. • ACL – Source IP address filtering to ensure that an access device is inserting a valid source IP address into headers. • DNS services – Domain Name System services. • DHCP services – Assignment of IP addresses and identification of other critical network assets. • VPN services – Network VPN technologies used to segregate traffic between two or more endpoints managed under a common subscription. • Packet/traffic anomaly detection/filtering.
Access Type	• Mobile device (2G – 5G). • Unlicensed service provider (e.g., Wi-Fi, CBRS). • Wired/optical (e.g., Ethernet, cable systems using DOCSIS, passive optical network (PON) technologies, and digital subscriber line (DSL) technologies).

Table 4.1.2.3 – WAN Functions and Use Cases

4.1.2.4 Application/Server Related Functions and Use Cases

FUNCTION/ATTRIBUTE	USE CASE – HOW THE FUNCTION IS USED
Server Type	• IoT application server • Device management server • Identity provider – Particularly for authentication/authorization purposes including

	support for session keys in P2P communication models, as well as delegation models. • Registration server – Enabling device registration for P2P applications, as well as NAT traversal services. • Network control servers (e.g., DNS, DHCP, CA - Certificate Authorities, SMS centers (SMSC)).
IoT Client Device	• Client device controlled by a user with access to the IoT service via an onboard application or through a web service.

Table 4.1.2.4 – Application Server Related Functions and Use Cases

4.1.2.5 Assets

An asset can be any object having value to someone: data, systems, devices, support infrastructure, people, structures, and facilities (e.g., operations centers or remote, unstaffed relay stations), corporate brand and reputation, and many other entities. An asset of any value is bound to be coveted by someone or targeted for mischief or malice. For the asset to be successfully attackable, it must have one or more characteristic that can be exploited or compromised in a way that will benefit the attacker, harm the asset user, or both. In this document, we focus our analysis on architectural assets including data, systems, functions, and devices.

Two types of assets are identified and defined as follows:

- **Primary assets** – Resources that represent the primary or fundamental value delivered by the IoT service or application.
- **Priority secondary assets** – Resources upon which the primary asset depends to the extent that they can be attacked with the intent of enabling the likely threat actors' goals associated with the primary asset.

For IoT services as discussed to the point, the primary assets are comprised of the IoT device and associated functions and data. Specifically:

- IoT device – Including the physical device (hardware) and associated operating system and support software.
- IoT device access port for connection services including connection specific credentials.
- IoT device management functions including management specific credentials to enable modification of code/data including:
    - Software/code
    - Configuration data
- IoT device operational functions including application specific credentials to enable access to:
    - Object control API
    - Data collection API
- Device compute functions including:
    - Physical processing
    - Memory/storage

Secondary assets include:

- LAN connecting the IoT device to the broader network.
- WAN connecting the LAN/device to servers and clients.
- Network and application servers including:
    - DNS
    - DHCP
    - CA (where applicable)
    - SMSC (where applicable)

- Client devices

4.1.3 Network Diagrams

The following architectural diagrams provide a framework to visually depict the physical and virtual system elements that make up the delivery of an IoT service or application, from network servers to end devices. The interrelationships and linkages of these architectural components can highlight potential vulnerabilities. The architectural diagrams also show the assets' location, both physical and logical. These assets have a topological and logical relationship with the use cases highlighted within the document.

Figure 4.2 illustrates the high-level network architecture of common IoT services provided over WAN networks. Each functional class described in section 4.1.2 is represented in this architecture with a summary of the key functions associated with each architectural element. As such, this network diagram can be used to illustrate each of the use case possibilities presented in section 4.1.2.

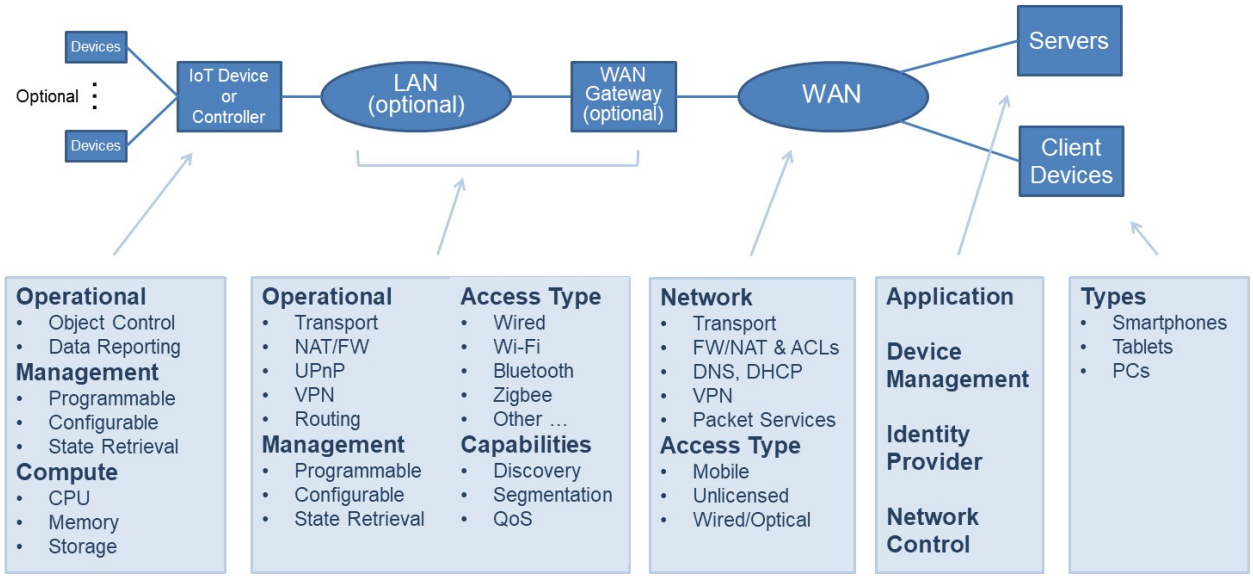


Figure 4.2 – IoT High-Level Architecture

The IoT service operates in several modes within the architecture including:

- **Network functional support** to provide the IoT device with a network-specific dynamic IP address, as well as other IP network configuration parameters such as the IP address of DNS services, and other network services as needed.

- **Network/device FW/NAT traversal** capabilities needed to enable network devices/servers to initiate the establishment of connections with the device. In many IoT architectures, it is necessary to have servers and devices with access to the WAN network be able to establish connections with the IoT device as needed. However, almost all practical WAN and LAN environments include FWs and potentially NAT devices, which prevent unsolicited device contact. As such, some means is necessary to enable clients and servers with access to the broad WAN to contact and establish connections with IoT devices while remaining within the policies of intermediate FW devices. Many different techniques and standards have been established to do this, each with their own inherent security attributes.

- **Device management** of the IoT device is, at a minimum, required to configure the device for the specific environment that it is located in.

- **Device operational functions** enable the IoT service to retrieve data from the IoT device (e.g., monitor function) or control specific capabilities within the IoT device environment (e.g., actuator functions).

The following network diagrams depict each of the IoT service modes listed above. In addition, trust boundaries indicate the demarcation between architectural components, which may be administered by different administrative domains. For example, the IoT device/controller is likely to be managed separately from the LAN it uses for connectivity. Similarly, the WAN operator may be independent of the LAN operator (at least from a management/administrative point of view). Communication that crosses trust boundaries (or between elements located in different trust domains) will require secure communication protocols to:

- Authenticate the point of contact in each trust boundary.
- Authorize use of resources.
- Ensure communication confidentiality, integrity, and availability.

The following sections discuss each of the four modes mentioned above, beginning with network functional support.

4.1.3.1 Network Functional Support

Figure 4.3 illustrates how the network may implement network functional support, as it is defined above, with emphasis on the configuration parameters.

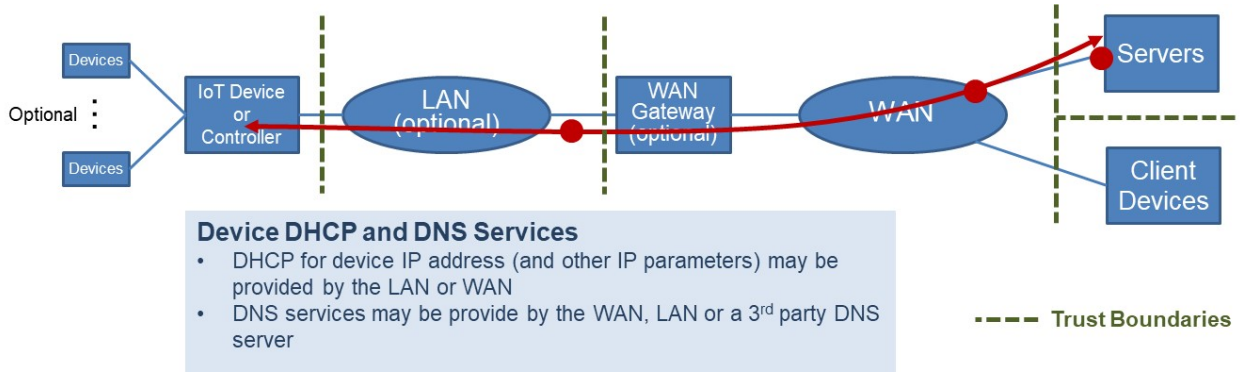


Figure 4.3 – Network Functional Support

When a device attaches to a network, the attach procedure invokes network services to acquire an IP address and other network-related configuration information. DHCP typically is used to provide IP configuration information to a device during attach. DHCP servers may exist throughout the network:

- In the LAN, particularly when the WAN gateway is using a NAT function.
- In the WAN to directly provide an IP address to the IoT device or WAN gateway where applicable.

However, DHCP is not the only way to configure the IP address of a device. Some WAN architectures use other mechanisms to convey IP address and server configuration settings. For example, the PDN gateway element in a 4G Enhanced Packet Core network uses mobility-specific control messages to assign dynamic IP addresses to mobile devices.

If needed, the attach procedure will also provide the IP address information used to contact a DNS server, which could be in the LAN, the WAN, or a data center connected to the WAN.

Once the IoT device has an IP address (and other needed IP network configuration parameters), the device can establish connections with other network and application elements. In order to inform the IoT application of its presence on the network, the device would typically register with one or more known IoT application

servers that have been pre-provisioned into the device. This allows IoT applications to discover and subsequently make use of the newly deployed device.

4.1.3.2 Network/Device Firewall/NAT Traversal

Almost all practical WAN and LAN environments include FWs and potentially NAT devices, which prevent unsolicited device contact by a network application server or P2P client device. As such, even when an IoT application server is aware of an IoT device's IP address (e.g., through the registration process discussed above), the IoT application server may not be able to contact the device without the aid of FW/NAT traversal functions. These functions enable connection establishment within the security policies defined in the intermediate FW/NAT devices.

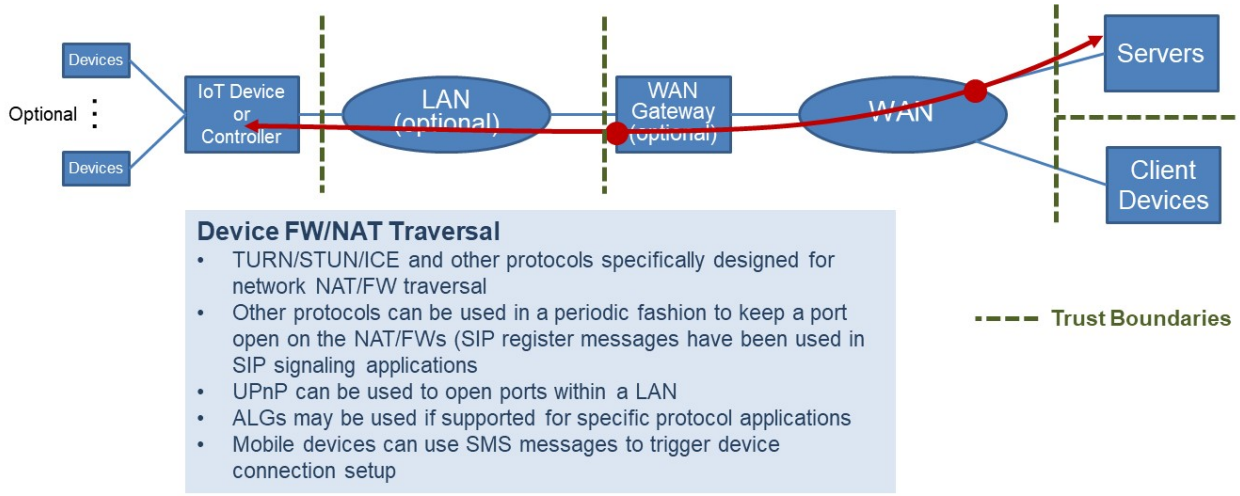


Figure 4.4 – Device/Network FW/NAT Traversal

Figure 4.4 illustrates options for the IoT device to establish a connection with an application server within the security policies of intermediate FW/NAT devices. Many options exist:

- The Internet Engineering Task Force (IETF) has defined a set of protocols that enable two-way communications between devices with intermediate FWs and NATs that otherwise block unsolicited traffic flows. Examples of such protocols include RFC 5389 – Session Traversal Utilities for NAT (STUN), RFC 5766 – Traversal Using Relays around NAT (TURN), and RFC 6544 – Interactive Connectivity Establishment (ICE). These protocols are commonly used for interactive media services such as voice over internet protocol (VoIP), P2P communications, video, and instant messaging applications, but they are applicable for IoT devices, as well.

- Depending upon the application, other protocols may also be used to keep a port open in FWs and NATs in the transport path. For example, SIP register messages have been used for this purpose in SIP signaling applications.

- In some cases, UPnP capabilities can be used to open a port in LAN gateway FW/NAT functions. However, UPnP does not address WAN -based FW/NAT functions commonly deployed in networks today, so this option might not always be feasible for IoT devices.

- In some cases, application layer gateways (ALGs) native to a FW/NAT device may provide the needed capabilities. Most FW/NAT devices include ALG functions to enable common VPN and file transfer protocols. Leveraging these build-in functions can simplify the FW/NAT traversal process by eliminating the need for additional signaling or server configurations.

- For mobile networks, the device may be in idle state with no active physical data plane connection. Mobile networks use control plane signaling (e.g., via trigged by an SMS message to the device) to cause the device to transition to an active state and contact the IoT application server. As such, an application server that needs to establish a connection with a mobile IoT device can stimulate the device via an SMS message to trigger the device to initiate a connection.

4.1.3.3 Device Management

Once the IoT device has registered and can be contacted by network-based application servers, remote management functions can be used.

IoT devices may be configured either through local or remote methods or both. Management functions include the capability to:

- Update the executable code on the device.
- Update configuration data, including security-related certificates or passwords.
- Access device state information related to the software or configuration state, provide location information, or upload performance metrics and logs associated with device operation.

Figure 4.5 illustrates typical device management functions applicable to IoT devices.

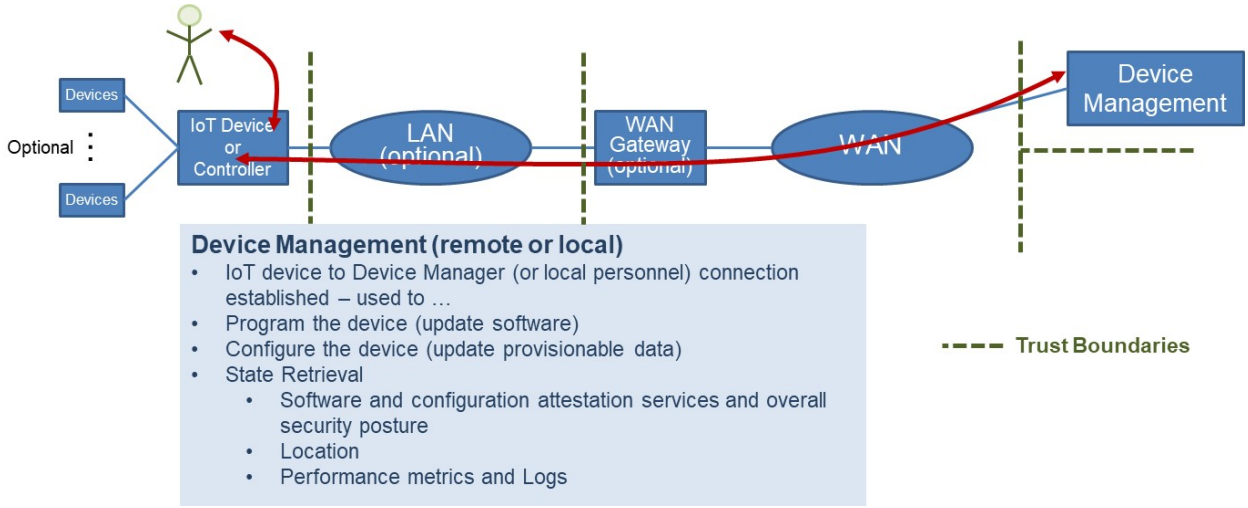


Figure 4.5 – Device Management Functions

4.1.3.4 Device Operational Functions

Finally, as shown in figure 4.6, the IoT device can be used with IoT services; providing control of objects and or report data associated with objects. The IoT device is typically accessed in two ways:

- The IoT device may connect directly with an IoT application server to exchange control or data information.
- The IoT device may connect with a client device.

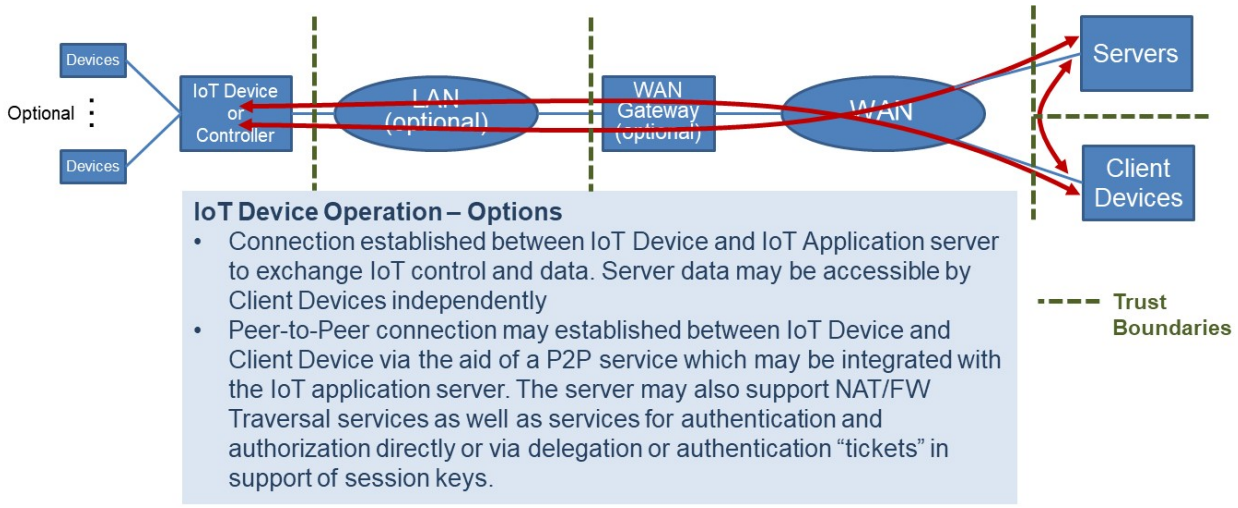


Figure 4.6 – Device Operational Function

4.1.3.5 Device IoT Application Layer Authentication

A key aspect related to IoT component connections is the ability to securely authenticate each end of the connection across the trust boundaries as shown in the previous sections. For IoT applications, two classes of authentication are commonly used:

- Shared secret – "Something you know."
- Trusted certificate – "Something you have."

Both methods are commonly used today, particularly for connections between IoT devices/clients and the IoT application servers.

For IoT P2P scenarios (typically when a client device communicates directly with an IoT device located on the premise), secure distribution of shared secrets to large populations of peers can be challenging. Rather than manage the distribution of large numbers of pair-wise secrets, "ticket"-based systems are commonly used to distribute session keys in real time as needed to the pair of devices that need to communicate. In ticket-based authentication systems, a device/client only needs a single shared secret to authenticate with the ticketing service rather than a separate secret for each possible client/device that it may need to communicate with.

A common example of a session key distribution system is Kerberos, developed in the 1980s at MIT. Kerberos (and other, similar mechanisms) rely on the following principles:

- Each client/device shares a unique secret with a centralized server.
- When two peers (e.g., an IoT device and a client device on the Internet) need to communicate, (i.e., initiate a session), the central server is invoked to generate a session key with an expiration time.
- The session key is used to encrypt parameters (called tickets) that can be passed between the communicating entities securely. This ticket provides the assurance to each entity that it is communicating securely with the intended party.

Authorization actions can be taken by the device/client or the central server as needed as messages are passed. In this way, the needed ticket is only passed for authorized connections.

Alternatively, per-device/client certificates can be used for authentication. Certificate-based systems can be:

- Private (self-signed or private CA).
- Public (trusted independent CA) using globally unique identifiers.

Certificate-based systems avoid the need for a centralized key distribution center for session keys as would be needed in ticket-based P2P systems. If entity A wishes to authenticate with entity B, A can forward to B its certificate. The certificate can be verified by B by checking the certificate signature. The certificate can be signed by a CA whose root certificate could be deployed to the devices/clients via the normal software update process.

Entity B may check a certificate revocation list in real time or one that has been cached. To distribute a session key, B can transmit the key to A by encrypting it using A's public key, which only A can decrypt with its private key. Mutual authentication can be achieved by repeating the above process in the opposite direction.

Certificates can be created and managed privately (with a self-signed certificate or private CA) or using a public, trusted, independent CA, with globally unique identifiers. Use of an independent CA has several advantages including:

- No need to create/support separate public key infrastructure (PKI).
- More controllable/visible level of security for the IoT application provider/user.
- Provides a consistent method for authentication; particularly for large enterprises that may utilize many different IoT applications across many different vendors.
- Provides support options if the vendor of a private PKI no longer offers that function.

In this section of the document, we have:

- Defined the primary security objectives for a wide class of IoT services.
- Documented key functionality and associated use cases that may be used in an end-to-end IoT application.
- Graphically represented the architecture and dataflows of key modes of operation for network-based IoT applications.

4.2 Threat Identification

Threat identification and the associated risk assessment steps are critical parts of the ARA process. Threat identification and the creation of a threat model aims to identify the attacks a system must resist and the defenses that will bring the system to a desired defensive state. The threat model itself, in order to assess risk, should include a weighting mechanism to quantify the risk associated with specific threats and to suggest where security controls should be placed to address the highest risk events.

There are many ways of performing threat modeling with a variety of industry-accepted methodologies and best processes. This variety allows organizations to adopt different processes or variations on these processes that best suit their needs. Because of this, we will not apply any specific methodology to the generic IoT system under investigation in this document. Rather, we provide insight into many of the key aspects integral to most threat modeling methods including:

1. Security objectives.
2. Identification of key assets, functions, and use cases exercising these assets/functions.
3. Network diagrams and information flows.
4. Attack classes, vectors, abuse cases, and potential mitigations.

Items 1-3 above have been covered in some detail in section 4.1. In section 4.2, we will focus on item 4: attack vectors, abuse cases, and potential mitigations. To do this, we first identify, for each asset, the specific attributes that can be attacked to undermine the asset in some way. We will then identify specific attacks and associated mitigations. Risk tends to be dependent on the specific IoT application, so it is not practical to accurately assign risks to the assets and associated attributes in a general way. That step must be done when the ARA process is applied to a specific service/application/system.

4.2.1 Identify Assets and Attributes

Each asset has a collection of characteristics, referred to as attributes, which are the set of defining qualities of the asset and consequently are the asset's attackable points. The set of attributes associated with an asset can vary depending upon the nature of the asset itself. For technology-related assets such as hardware modules, software functions, and services, it is useful to leverage the set of attributes commonly used in the context of security. These standard attributes are:

- **Availability** – Ensuring timely and reliable authorized access to and use of an uncompromised asset.
- **Authenticity** – Reflecting the property of being verifiably genuine (i.e., associated with the identity as claimed by the asset and/or an independent root of trust).
- **Confidentiality** – Ensuring that the asset is protected from unauthorized use (i.e., the asset can only be disclosed to or used by authorized, authenticated identities).
- **Integrity** – Ensuring that the asset has not been modified by unauthorized identities and thus operates consistent with non-altered instances of the asset.

In an ARA analysis, it is important to use a complete set of consistent attributes for each IoT asset. Specifically, for a given asset, the attributes used should be clearly defined to cover all anticipated attack effects without overlap. By overlap, we mean each identified attribute is orthogonal in the sense that attacks on a given attribute do not bleed over to other attributes. It should be noted that the attribute definitions listed above are chosen to ensure both completeness and orthogonality.

Table 4.1.1 – Assets and their Attributes

Asset	Attributes
IoT Device	Availability, Authenticity, Confidentiality, Integrity
IoT Device access port for connection services	Availability, Authenticity, Confidentiality, Integrity
IoT Device management functions • Software/code • Configuration data	Availability, Authenticity, Confidentiality, Integrity
IoT Device operational functions • Object control API • Data collection API	Availability, Authenticity, Confidentiality, Integrity
Device compute • Physical processing • Memory / storage	Resource Availability, Integrity
LAN	Availability, Authenticity, Confidentiality, Integrity
WAN	Availability, Authenticity, Confidentiality, Integrity

Network and Application servers • DNS • DHCP • CA (where applicable) • SMSC (where applicable)	Availability, Authenticity, Confidentiality, Integrity
Client devices	Authenticity, Confidentiality, Integrity

Each asset/attribute pair noted above is exposed to an ongoing potential for harm from an attack or class of attacks. The harm is not realized until an actual attack occurs and, as such, we consider the potential harm rather than actual harm. The various types of exposure to harm can be categorized and we will refer to these categories as “exposures” in our subsequent analysis.

For example, the confidentiality attribute of data is subject to attacks that result in data disclosure or data theft. Therefore, theft and disclosure are two exposures associated with data confidentiality. The actual loss of confidentiality is realized only through a successful attack; the exposure to potential harm is always there.

Typical exposures associated with IoT systems include:

- Denial of service.
- Unauthorized data disclosure or theft.
- False data.
- Tampering – intentional modification of an asset in a way that would make them harmful.
- Misuse of service – using the existing service in a way that would do harm, such as remotely turning the heat off in a remote location.
- Refutation – commonly associated with non-repudiation and the denial of a claim related to use of or data associated with an asset.

4.2.2 Attack Classes, Vectors, and Mitigations

Attack classes are general groupings of attacks having common characteristics. Grouping attacks into classes benefits the analysis, quite often a specific mitigation can be used to address all or a large segment of the entire class. Thus, by identifying attack classes rather than specific attacks, we can establish a set of defenses that address broad categories of attack; which brings a more strategic approach to security. Attack classes are general rather than specific. Attack classes can be further expanded into countless subcategories, types, and varieties of attack.

Commonly seen IoT related attack classes include:

- **Physical/mechanical tampering or destruction** – This represent attacks associated with a physical presence, attacking the environment (electrical power, HVAC, or physical destruction) of the IoT device, as well as manual tampering of the device to stop it from operating immediately or perhaps later. Some physical attacks may require a physical presence by the attacker or agent, though some attacks—such as explosive attacks or gun shots—can be delivered from some distance away.

- **Software attacks** – These are a wide range of attacks that target the software image of IoT devices and servers. By compromising the software (code and/or configuration data) of an IoT device or IoT server, the attacker can take control of the device/server to:
    - Steal data (e.g., from IoT sensors).
    - Take control of the physical equipment associated with an IoT device (control an actuator).

- Disable the IoT device/server.
- Launch DDoS attacks on other servers/applications.
- Propagate malware to other devices (IoT or otherwise).

This class of attacks can utilize a variety of entry points:

- Insider attacks that physically compromises IoT devices, controllers or servers thereby affecting code and or configuration data.
- Exploits associated with software vulnerabilities that allow code insertion and/or modification of configuration data.
- Credential theft to enable the attacker to use standard maintenance interfaces to modify code or configuration data.
- Supply chain compromise, which, for example, may enable a back door for future malicious intent (not explicitly addressed in this analysis).

- **Network-centric attacks** – These leverage the networked nature of IoT systems to compromise or otherwise subvert proper system behavior. Fundamentally, these attacks endeavor to compromise the communications integrity of the systems. Examples include:

  - Preventing access to an IoT component though DDoS attacks or other means that disable the device. For example, a network-based attack could deplete the batteries of battery-operated devices in order to make the device prematurely inoperable.
  - Snooping to gain access to IoT traffic for the purpose of stealing data and/or credentials.
  - Man-in-the-middle attacks to compromise the integrity of the IoT device communications potentially leading to data theft, control failures, or service inoperability.
  - Routing-based attacks that may redirect packets to a different location for malicious purposes.

- **Credential and identity attacks** – These are enabled by the attacker to either insert new (malicious) components into the IoT system or gain "authorized" access to functions that may then be used for malicious purposes. This type of attack includes:

  - Spoofing attacks to insert malicious devices into the network for the purpose of disrupting proper functioning of the IoT system.
  - Credential theft or brute force credential acquisition that allow the attacker to launch software attacks (see above) and/or gain access to operation and maintenance functions to enable data theft, malicious device control actions or software/configuration modifications.

- **Data attacks** – These target data associated with an IoT device and may include:
  - State data associated with a device (e.g., location of the device).
  - Events being monitored by the IoT device (e.g., seismic activity, water levels, temperature).
  - Control data used to control the device (e.g., the ability to actuate a specific IoT control point with specific data).

4.2.2.1 IoT Device Assets

The IoT device and associated functions and data comprise the primary assets in the IoT risk analysis. The IoT device is the center point of data collection and object control functions used in IoT services and applications.

Table 4.2.2.1 provides a starting point for the IoT device threat analysis by listing typical attacks within each attack class noted above and potential mitigations. This table is not intended to be comprehensive but should form a solid basis for the analysis.

Because we have chosen attack classes that are aligned with assets, it becomes easier to identify mitigations which are typically created to protect a specific asset or function. The potential mitigations in

the right-hand column of the table provides a head start on the “next step” once the high-risk assets have been identified.

Table 4.2.2.1

ASSET	ATTACK CLASS/ATTACKS	MITIGATIONS
IoT DEVICE		
	Physical • Physical device destruction • HVAC tampering to facilitate device malfunction • Electrical system tampering to affect device operation • Fire suppression system tampering in order to cause flooding or prevent fire suppression • Tampering with the physical device and/or location of the device **Software/Configuration:** • Acquire and reverse engineer to uncover vulnerabilities • Gain physical access to attack site to insert vulnerability in device • Supply chain compromise to insert surreptitious code **Credentials**: • Supply chain compromise to compromise device in many ways	• Use of physical security mechanisms and procedures • Use of HW root of trust technology using tamper resistant hardware (e.g., Trusted Platform Module (TPM), Trusted Execution Environment (TEE), Universal Integrated Circuit Card (UICC) • Physical access security procedures • Secure supply chain processes and testing
ACCESS		
Access (L2) Link	**Network / Software**: • Software vulnerability exploited through network access to enable remote	• Robust and timely software update policies and procedures

	code execution for control of maintenance and/or operations functions	
Access (L3+) Network	**Network/Software**: • Software vulnerability exploited through network access to enable remote code execution for control of maintenance and/or operations functions	• Robust and timely software update policies and procedures
Authentication and Authorization (all layers) including access credentials	**Credentials** • Credential theft. • Credential compromise due to (say) default or weak credentials	• Separate credentials for access, management and operational functions • Strong credentials • Mutual authentication • Multi-layer authentication • Context based authentication (location, user profile, date/time, network state) • HW root of trust (TPM, TEE, UICC, …) • Potential need for standards to address IoT device delegation and peering models
TLS support (or similar)	**Credentials**: • Man-in-the-middle root certificate exploit • Certificate management exploits	• Pinning root certificate • Employ HW root of trust (e.g., TPM, TEE, UICC) for certificate store/management
Delegation and peering support for authentication and/or authorization	**Credentials:** • Credential compromise due to default or weak credentials	• Enforce strong credentials. • Potential need for standards to address IoT device delegation and peering models

MANAGEMENT FUNCTIONS		
Programmable	**Software:** • Device code compromised	**Prevention**: • Secure software update procedures • Test against set of known software attacks **Detection**: • Anti-virus software • Audits on hash of the code • Signed code hashes • Remote attestation via TEE/TPM technology
Configurable	**Software:** • Device configuration compromised	**Prevention**: • Real time checks for valid configurations **Detection**: • Remote audits for valid configurations • Remote attestation of configuration via TEE/TPM technology
State Retrieval	**Data:** • Theft of data associated with device state (e.g., location)	• Utilize data encryption technologies
Management Credentials	**Credentials:** • Credential compromise due to default or weak credentials	• See credential-based mitigations
OPERATIONS FUNCTIONS		
Service – object control	**Data:** • Unintended control (non-operative or destructive control) through insertion of control data for malicious purposes	• Control data audits for valid values • External/redundant verification/protection devices

Data – monitor object	**Data:** • Theft of data being monitored by the device	• Redundant measurements potentially using different IoT devices • Utilize data encryption technologies
Application credentials	**Credentials:** • Credential compromise due to default or weak credentials	• See credential-based mitigations
INTERNAL (COMPUTE)	**Network:** • Network-based attack designed to exhaust internal compute resources	• Software overload mechanisms to minimize impact

4.2.2.2 LAN Assets

The LAN functions serve to connect the IoT device to other elements in a network to form an IoT system. The LAN component exists when the IoT device is not directly connected to a WAN (as might occur for cellular IoT devices).

The LAN functions can be considered secondary assets (see section 4.1.2.5) within an IoT networked system because attacks on the LAN are likely to affect a wide variety of other services, while IoT-impacting attacks are more likely to be narrow in scope. LAN-based IoT attacks are most likely to impact IoT availability because they do not touch the IoT device directly. For instance, a LAN attack may be constructed to cut off communication to the IoT devices, thereby affecting device availability but having no effect on integrity or any other IoT device attribute. LAN vulnerabilities may also be used to assault IoT "data in motion," resulting in theft of data (which is an issue when the asset is the data itself). Fortunately, this attack is generally mitigated through data encryption techniques.

The LAN itself can provide added security protection for IoT devices in several ways. For example:

- LAN elements generally include a WAN gateway function, which commonly includes a FWl providing some level of protection to LAN devices.
- LAN capabilities often include the ability to segregate traffic (e.g., in a VLAN). Traffic segregation is an excellent mechanism to limit network-based attacks because it limits the exposure of LAN devices.

Table 4.2.2.2 provides a starting point for the IoT LAN threat analysis by listing typical attacks within each attack class noted above and potential mitigations. This table is not intended to be comprehensive but should form a solid basis for the analysis.

Table 4.2.2.2

FUNCTION / ATTRIBUTE		
WAN GATEWAY	**Network:** • Internet based port attacks from anywhere in the world • Attack on WAN maintenance interface (remote) • Attack on LAN maintenance interface (local) • Network based attack designed to exhaust compute/memory and/or switching network resources **Software/Configuration:** • Code or configuration compromised	• FW/NAT (IPv4) • FW (IPv6) • Strong and unique Pre-Shared Key (PSK) or • gateway specific certificate. • Maintenance interface segregation • Software overload mechanisms to minimize impact **Prevention:** • Secure software update procedures • Known software attack testing and prevention **Detection:** • Anti-virus software • Audits on hash of the code or configuration • Signed code hashes • Remote attestation via TEE/TPM technology
IoT CONTROLLER	Same as 4.2.2.1 IoT Device	Same as 4.2.2.1 IoT Device
LAN ACCESS TYPE	**Network:**	
Wired	• Physical compromise of wired network	• Physical access security procedures • Use of wired authentication methods such as 802.1x
Wi-Fi	• No or weak Wi-Fi security (e.g., open or	• Use Wi-Fi Protected Access II (WPA2) or

	Wired Equivalent Privacy (WEP))	802.1x methods with strong credentials
Zigbee	• Zigbee access security vulnerabilities	• See Zigbee security documentation
Bluetooth	• Bluetooth access security vulnerabilities	• See Bluetooth security documentation
LAN CAPABILITIES	**Network:**	
Transport	• Snooping • Man-in-the-middle • DDoS	• Physical access security procedures • Strong L3 security e.g., TLS
UPnP	• Malware highjacks UPnP to open ports for malicious purposes	• Use of alternative methods to open ports
Network Segmentation	• Compromised devices affect other devices on the LAN	• Use of VLANs to segregate classes of devices • Use of DMZ with FWs to segregate IoT components

4.2.2.3 WAN Assets

The WAN functions serve to connect the IoT device to other elements in a WAN network to form an IoT system.

Like their LAN counterparts, WAN functions are secondary assets within an IoT networked system. Attacks on the WAN are likely to affect a wide variety of other services, while IoT-impacting attacks are more likely to be narrow in scope. WAN-based IoT attacks are most likely to impact IoT availability by inhibiting communication to the IoT devices. WAN vulnerabilities may also be used to assault IoT data in motion for theft of data attacks, although this kind of attack is generally mitigated through data encryption techniques.

The WAN itself can provide added security protection for IoT devices in several ways. For example:

- Some WAN access technologies (such as 3GPP mobile access) deploy a FW in the WAN access network providing some level of protection to WAN devices.
- Many WAN services exist to enable IoT traffic segregation. For example, WAN VPN technologies can be used to separate IoT traffic from other (e.g., Internet) traffic between the IoT devices and the IoT servers in a data center.

Table 4.2.2.3 provides a starting point for the IoT WAN threat analysis by listing typical attacks within each attack class noted above and potential mitigations. This table is not intended to be comprehensive but to form a basis for the analysis.

Table 4.2.2.3

FUNCTION/ATTRIBUTE		
Network Transport	**Network:** • Snooping • Man-in-the-middle	• Network NAT/FW

	• DDoS	• Access control lists on source IP address • Protocol anomaly filtering • Service provider DDoS services • VPNs and other network segmentation technologies
Access Type	**Network:** • Known 2G attacks	• Avoid use of 2G technologies

4.2.2.4 Network and Application Server-Related Assets

The application server functions provide the application processing and storage needed to implement IoT applications based on the data and control capabilities of IoT devices.

These functions are secondary assets within an IoT networked system because attacks on the servers are generally common to a wide variety of other services. General data center and server security mechanisms should be deployed to protect data center services.

In some cases, network servers may have internet-accessible account access. For example, by using compromised credentials to gain access to the authoritative DNS server, an attacker can modify the location to which an organization's domain name resources resolve. This lets the attacker redirect user traffic to attacker-controlled infrastructure and obtain valid encryption certificates for an organization's domain names, thus enabling man-in-the-middle attacks. To mitigate attacks such as these, best practices should be used in securing account access to network services. These include:

- Regularly updating all accounts that can change network service parameters with strong passwords.
- Implementing multifactor authentication on internet accessible accounts, or on other systems used to modify network related records.
- Audit public DNS records to verify they are resolving to the intended location.

4.2.3 Identify Abuse Cases and Rank Threats

The last step in the threat identification phase of the ARA process is to apply pruning and weighting methods to prioritize threats and identify abuse cases to quantify the impact of potential attacks. Abuse cases are a combination of use cases and attack vectors and can generally be derived from the tables listed in section 4.2.2.

The end goal of this step of the ARA is to create a ranked list of threats correlated to assets and interfaces. This information can then be used as input into a risk analysis and threat mitigation plan. Ideally, the threat modeling method used in the analysis should provide this ranked list of threats.

For example, a tree structured threat model mechanism could be applied where:

- For each identified asset, rank the attributes that are associated with that asset (see table 4.2.1) by importance. For example, the IoT device asset has availability, authenticity, confidentiality, and integrity as attributes. Depending on the specific IoT application being analyzed, one might rank the attributes as follows:

Table 4.2.3

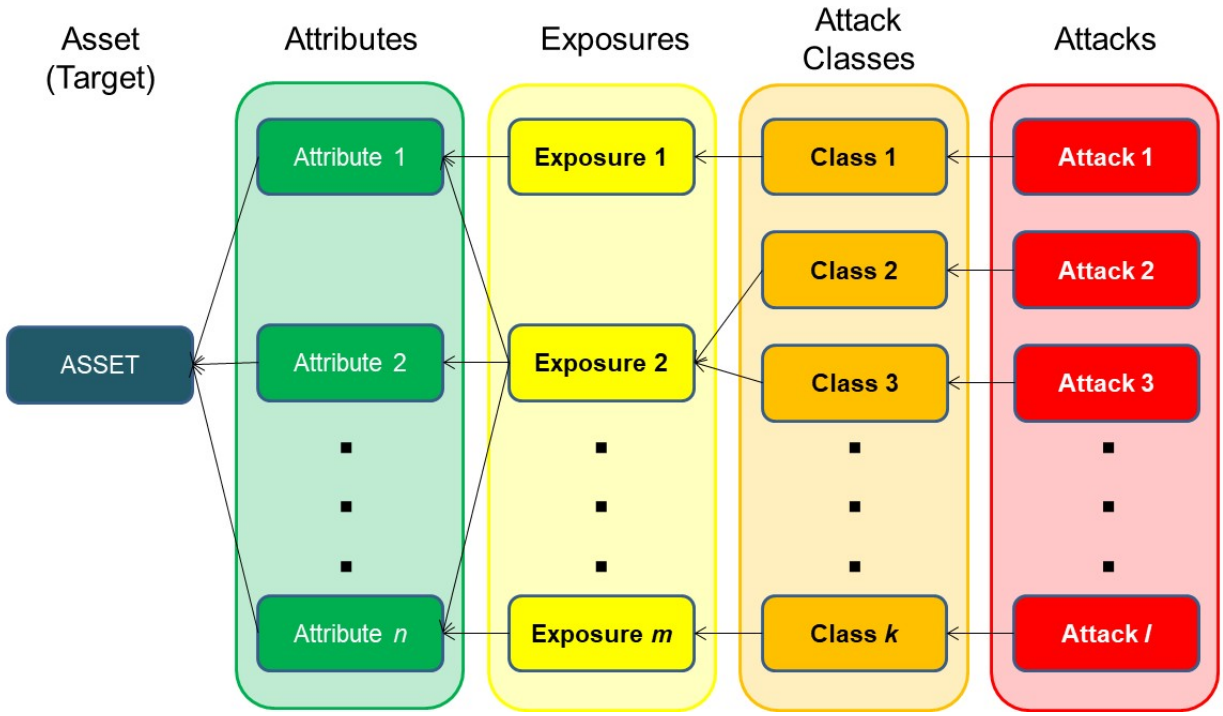
ATTRIBUTE:	RANK:
Availability – for example, many redundant sensors have been deployed and as such, loss of availability of a small number of devices may not be critical.	1 – Least important
Confidentiality – for example, the actual sensed data from a specific IoT device sensor may not severely compromise the IoT system.	2
Authenticity – for example, it is important that the sensor identity may be bound to location or other critical aspects that are important to the IoT service being provided.	3
Integrity – for example, it may be critical that the sensed data be accurate.	4 – Most important

Different IoT services and applications may have much different needs and associated rankings.

- For each asset, one can start to create a tree structure where the first column of nodes maps to the ranked attributes.
- The next column of nodes in the tree consists of exposures associated with each attribute. Exposures can be directed to more than one attribute. An exposure refers to a category of potential exposures to harm from an attack or class of attacks. Typical exposures may include:
    - Denial of service
    - Unauthorized data Disclosure or theft
    - False data
    - Tampering
    - Misuse of service
    - Refutation
- The next column of nodes in the tree represents attack classes, which are mapped to each exposure. Attack classes can be directed to more than one exposure.
- As needed, map specific attacks to the attack classes identified.
- Adversary assumptions can be used to prune the target tree.

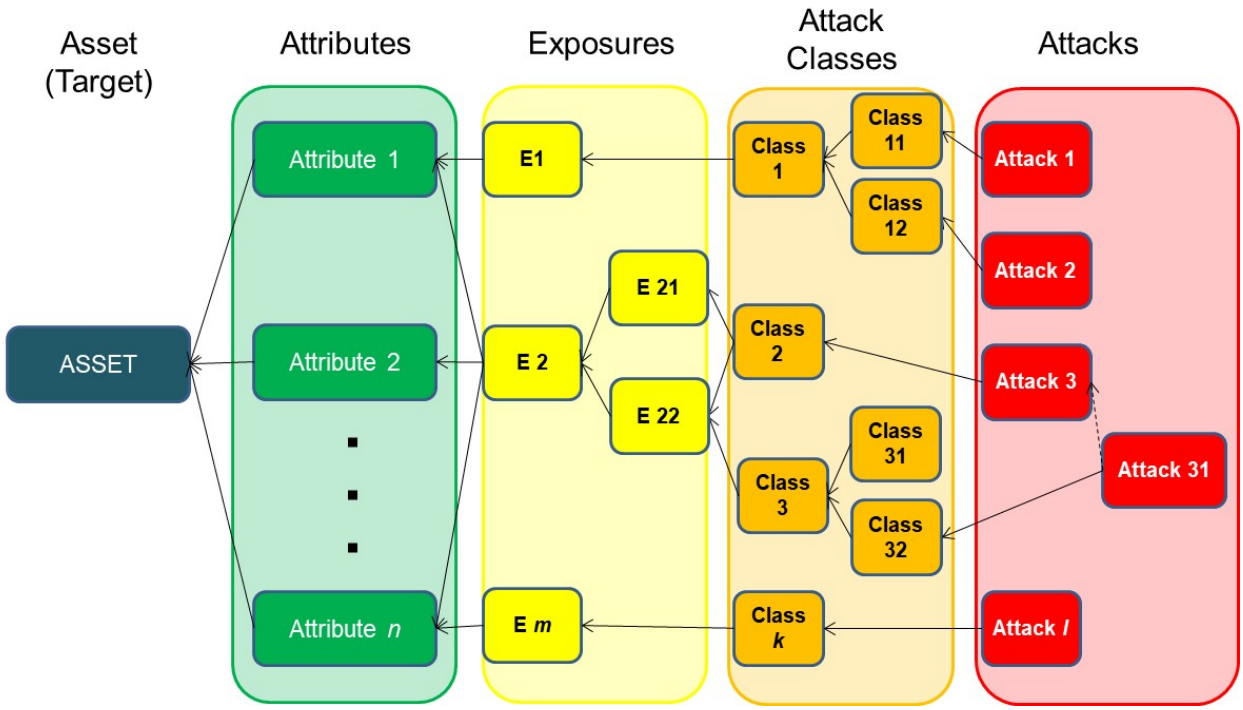
Figure 4.2.3 illustrates the tree structured threat model as described above.

Figure 4.2.3



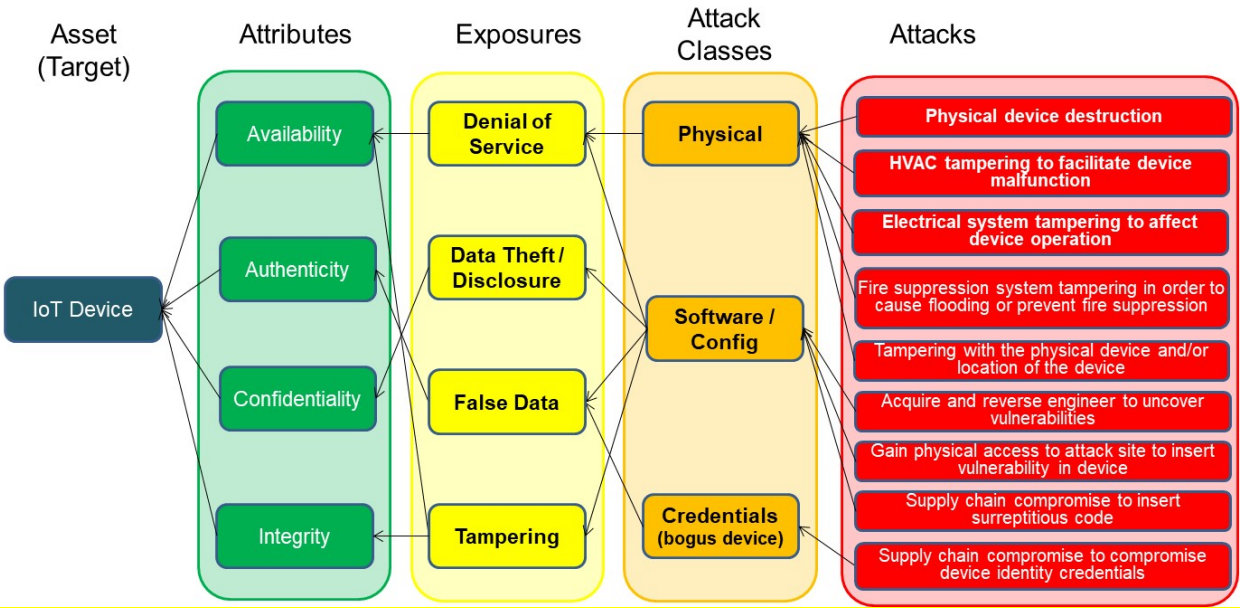
It is important to note that in some cases, when identifying exposures, attack classes and attacks, it is possible that dependencies exist within a category. For example, an identified exposure may be dependent on two or more other exposures. Similarly, you may identify attack classes or attacks that are dependent on two or more other attack classes or attacks. This dependency can be shown in the tree structure as illustrated in the Figure 4.2.4.

Figure 4.2.4



We can apply this threat modeling technique to the generic IoT system described in this document. Table 4.2.1 lists the attributes associated with each asset that has been identified. Thus, for each asset listed in this table, we have the set of attributes which comprise the first column of our tree structured model. Tables 4.2.2.1 through 4.2.2.3 provide tables that enumerate potential attack classes and attacks for each identified asset. This information can be directly applied to the attack class and attacks column for the threat model. Common exposures have been enumerated at the end of section 4.2.1. This information can be used to populate the exposures column. The result is illustrated in the following figure for the first identified asset, the IoT device (including the physical device hardware and associated operating system and support software).

Figure 4.2.5



As shown above in figure 4.2.5, when creating this tree structured model, we can create a weight metric for each node in the tree by adding the weights of the attributes nodes as we traverse the target tree, from the attribute nodes to the leaf nodes (i.e., from left to right) through the target tree. Because this metric is based on the ranking of the asset's attributes, the weight calculated for each attack can indicate the most dangerous threats to the asset, to a first approximation.

We can also create a path metric that is based on the number of nodes and paths traversed in tracing each attack (or attack class) to the attributes it endangers. For each node, the path metric is calculated by adding the initial path values of the attack and attack classes leaf nodes, moving from the leaf nodes to the attributes (from right to left in figures) through the target tree. The leaf nodes representing the attacks or attack classes are arbitrarily given an initial value of 1.

These metrics help us build a ranked list of attacks by identifying which attacks affect each attribute of an asset. In addition, we can identify the most attackable (i.e., vulnerable) attributes for each asset.

Figure 4.2.6

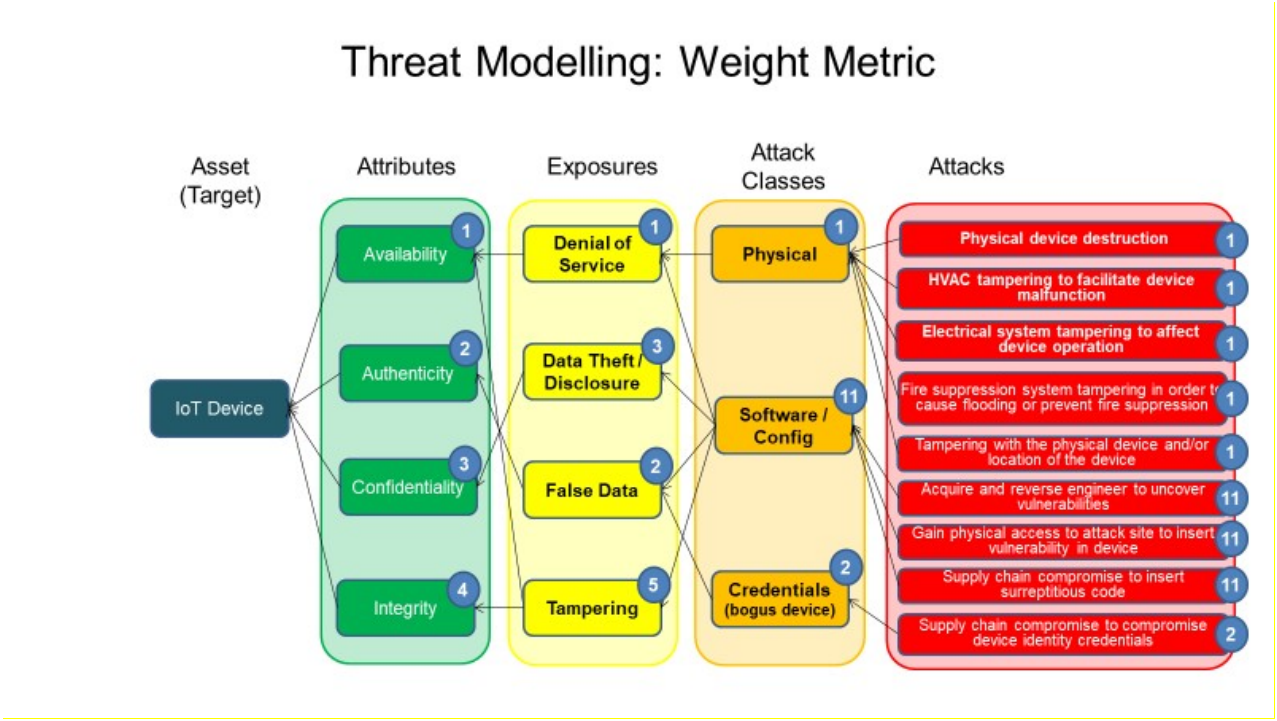


Figure 4.2.6 shows the weight metrics for our example threat model diagram. Attribute priorities listed are based on the analysis in Table 4.2.3. In this example, software/configuration-based attacks carry the highest weights pointing to the need to look closely at how IoT device software and configuration information can be adequately protected from tampering attacks, reverse-engineering attacks, and insertion attacks resulting from physical access to the device.

Figure 4.2.7

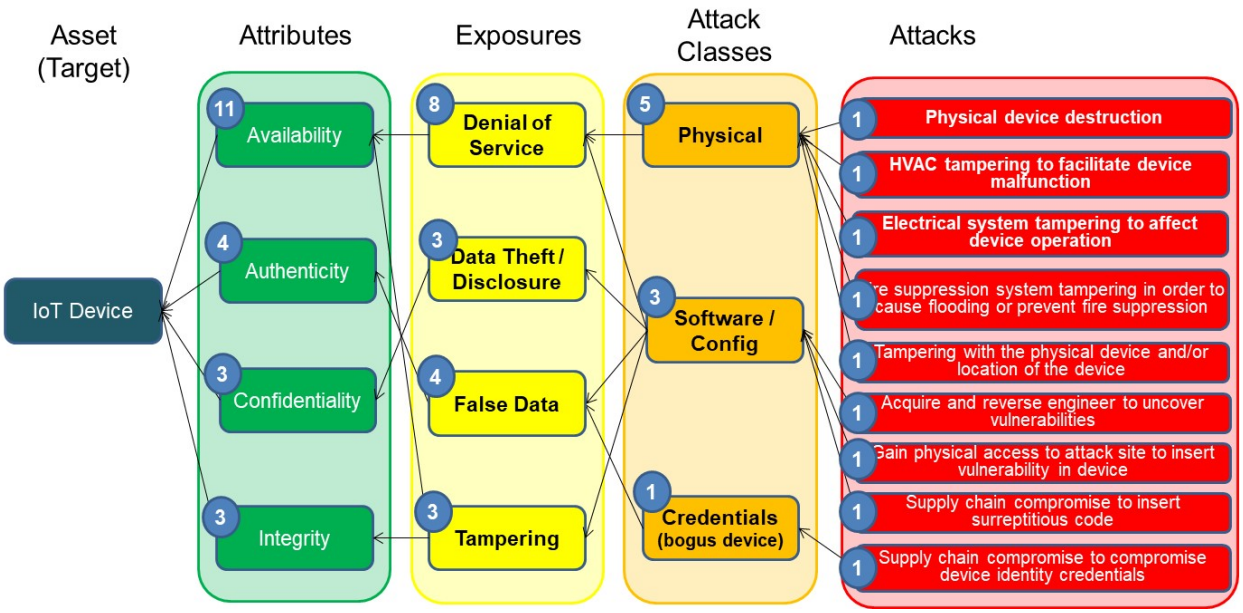


Figure 4.2.7 shows the path metrics for our example threat model diagram. In this example, we see that availability is the most attackable attribute, as the path matric of 11 indicates. Advantageously, availability is also the lowest priority attribute (as indicated by a weight metric of 1) based on our prioritization analysis in Table 4.2.3. Because many redundant sensors can be deployed and, as such, loss of availability of a small number of devices may not be critical to overall system operation.

4.2.4 The Adversary-Centric Aspect of the ARA

The adversary-centric aspect of the ARA considers the threat agents who would attack the asset. Lists of threat agents have been compiled in various threat agent collections: compendiums of threat agents defined and produced by various companies and organizations. The ARA relies on the one produced by Intel.[7] Referred to as the Threat Agent Library (TAL), this compendium provides 21 categories of threat agent, each uniquely identified by a combination of defining attributes.

A valuable TAL characteristic is its recognition that all adversaries are constrained by their skill level, their access to the asset, the resources at their disposal, and other important factors referred to as defining attributes that every threat agent possesses to some degree. Table 4.2.4 lists those attributes, along with categories that characterize each. The defining attributes help us to determine which of the TAL's 21 threat agents pose potential threats to the asset and to eliminate irrelevant threat agents by discounting those whose motives, access to the target, skill needed to affect an attack, etc., do not qualify them as credible or capable attackers. Once we have identified that threat agents we need to worry about, we are able to identify which attacks in the target tree each threat agent can launch.

Table 4.2.4: Threat Agent Attributes

	Defining Attribute	Category
1	Intent	Hostile or non-hostile
2	Access	Internal or external
3	Desired outcome of attack	Acquisition or theft, business advantage, damage, embarrassment (to attacked party), technical advantage.
4	Limits (legal and ethical limits that may constrain the agent)	Code of conduct, legal, minor extra-legal, major extra-legal.
5	Resources	Individual, club, contest, team, organization, government.
6	Skill level	None, minimal, operational, adept.
7	Objectives	Copy, deny, destroy, damage, take, all/don't care.
8	Visibility	Overt, covert, clandestine, multiple/don't care.

To carry out the adversary-centric portion of the analysis, we use the TAL to determine the relevant threat agents. We then examine each threat agent in turn to decide which attacks and attack classes in the target tree the threat agent is likely to carry out. We prune the target tree by removing all attacks and attack classes not relevant to that threat agent. This activity produces a smaller, pruned tree that identifies the

[7] Casey, T. (2007, September). "Threat Agent Library Helps Identify Information Security Risks." Intel white paper.

threats posed by that threat agent. We then apply the Weight and Path metrics to quantify the risk from that threat agent. We repeat the process for each threat agent in our list. Completing this exercise gives us the threat agents plus an estimate (via the metrics) of the seriousness of the overall threat each one poses.

It should be recognized that part of the relevance of each threat agent to the target asset stems from the nature of the asset itself. If the target asset is a small, unknown, and unimportant application, attacks might be more likely to originate with less lethal threat agents. If, however, the target asset's status was that of a national infrastructure, it might find itself a target of more serious contenders such as organized crime and nation-states.

By identifying the asset and its value, the threats to the asset, and the adversaries likely to carry out those threats, we are positioned to evaluate the risk to the asset and identify the correct defenses we must mount to protect it from the recognized threats.

4.3 Risk Analysis and Threat Mitigation Plan

The objective of a risk analysis and mitigation activity is to assess the risk to an asset from one or more threats. The next step is determining the nature and types of the defenses that must be mounted to protect the asset from the perceived threats, followed by assessing the effectiveness of the security controls and countermeasures that constitute those defenses. The intent is to identify where to further strengthen security or to identify gaps that need additional security controls applied.

To complete this analysis, a threat model (similar to the example shown in 4.2) should be created for each asset listed in table 4.2.1. The specifics of each threat model should be based on the adversary centric aspects discussed in section 4.2.4. This may result in many different treat models for many different assets. By prioritizing both assets and adversaries, an attack profile can be created, which can be used to assess risk and to create a robust security mitigation plan for the IoT solution under analysis. The tables listed in section 4.2.2 provide potential security mitigation mechanisms that could be applied to the solution. Ultimately, the mitigation plan (which is based on the risk assessment and is itself outside the scope of this paper) should include the set of security mitigations that best protect the most critical attributes of the most critical assets of the end-to-end solution.

5 IoT Security Conclusions and Additional Work

This document offers a robust analysis of a representative IoT end-to-end system. In this analysis, we leverage the ARA process developed and documented by ATIS in *Cybersecurity Architectural Risk Analysis Process.*[8] The following steps were carried out for this analysis:

1. **Security objectives** – Security objectives were documented that correlate to the functionality and services provided in an end-to-end IoT system.
2. **Use cases** – Key functions of an IoT system were identified and described. Assets and their attributes associated with these functions are listed.
3. **Network diagrams** – Architectural diagrams showing the major assets, functions, interfaces, and trust boundaries of the solution were developed. Data flows, trust boundaries, and protocols of critical data paths and operational modes were identified.
4. **Assets, risks, and threats** – Assets and attributes associated with the key functions and modes of operations were identified along with, threats, threat vectors, and potential mitigations.
5. **Risk analysis and threat mitigation plan** – Guidance was provided on how to use the information presented to develop a robust risk analysis and threat mitigation plan.

We observe that IoT systems share many common aspects that differentiate these systems from non-IoT networked systems. For example:

[8] ATIS. *Cybersecurity Architectural Risk Analysis Process*. ATIS-I-0000057. May 2017.

- IoT systems generally consist of a set of IoT devices networked via LAN and/or WAN networking functions to IoT servers and other IoT clients.
- IoT devices have two primary operational functions: to monitor and report on some aspect of the environment that they are deployed in and optionally, and to control aspects of objects located in the environment.
- IoT devices are often “simple” low-power devices with limited compute and memory/storage resources, plus specialized software to execute dedicated functions.
- IoT devices may not have a direct human-to-machine interface. When they do, allowable operations tend to be limited to basic configuration or the operational functions associated with the device. As such, security vulnerabilities that rely on deceiving a human operator are of less concern.
- IoT device software images tend to be more controlled because the IoT device is generally not dependent on other application code loaded by a human.
- The potentially large set of IoT devices deployed in an environment presents opportunities for redundancy but may also complicate authentication and the management of device credentials in a scalable fashion.
- Access to IoT devices may require use of P2P technologies, creating new risks associated with these technologies.

The effort to produce this IoT system risk assessment reveals a level of complexity that complicates the creation of a robust security strategy. For example, a typical IoT system includes:

- Many different components including a wide variety of IoT devices, networking capabilities, servers, and client devices.
- Situations in which each of these components may support many key IoT-related assets across many different operational and management modes.
- Situations in which each of the components and their associated functions may be key assets necessary for a working IoT system.
- Each asset possibly having many attributes that must be secured.
- Attributes that may be attacked directly or indirectly in vector-based attacks in a wide variety of ways.
- Attacks that might be intensified by the capabilities and attributes of the assumed adversary.

Despite the difficulties encountered while producing this risk assessment, applying the ARA methodology to the generic IoT asset has provided a reasonable framework for managing IoT risk assessments. This framework solidifies the process of:

- Identifying primary, secondary, and other IoT real-world assets.
- Establishing a complete set of consistent and useful attributes for each IoT asset.
- Providing a reasonable subset of attack classes to each of those attributes.

The exercise of being able to produce and work through a simple though plausible example of a generic IoT asset illustrates that the ARA can be applied to an IoT device or service once the set of attack classes has been identified and the initial sets of metrics has been applied.

In this environment, articulating and prioritizing attacks and attack classes given all these considerations can be difficult and could benefit from automated tools and formal methods to assist in the analysis. Future work to enhance and automate the ARA methodology is a worthwhile undertaking that could provide an alternative to the current IT risk assessment methods, provide a more comprehensive and revealing process than most existing methods offer, and establish a new baseline for risk analysis at large.

6 Bibliography & References

ATIS. *Cybersecurity Architectural Risk Analysis Process*. ATIS-I-0000057. May 2017. Alliance for Telecommunications Industry Solutions; Washington, DC.

Casey, T. (2007, September). "Threat Agent Library Helps Identify Information Security Risks." Intel white paper.

7 Glossary

3GPP	3 Generation Partnership Project
4G	4th Generation Wireless Mobile Technology
5G	5th Generation Wireless Mobile Technology
ACL	Access Control List
ALG	Application Layer Gateway
ANSI	American National Standards Institute
ARA	Architectural Risk Analysis
API	Applications Programming Interface
ATIS	Alliance for Telecommunications Industry Solutions
Bluetooth	A wireless technology standard for exchanging data over short distances
CA	Certificate Authority
CBRS	Citizens Broadband Radio Service
CIA	Confidentiality, Integrity and Availability
CITEL	Inter-American Telecommunication Commission
CPU	Central Processing Unit
DDoS	Distributed Denial of Service
DHCP	Dynamic Host Configuration Protocol
DMZ	Demilitarized Zone – An isolated network positioned between the Internet and a private network.
DOCSIS	Data Over Cable Service Interface Specifications
DNS	Domain Name System
DSL	Digital Subscriber Line
FCAPS	Fault, Configuration, Accounting, Performance and Security
HVAC	Heating, Ventilation, and Air Conditioning
HW	Hardware
ICT	Information and Communications Technology
IETF	Internet Engineering Task Force
IoT	Internet of Things
IP	Internet Protocol
ITU	International Telecommunication Union
LAN	Local Area Network
LoRa	A patented Long Range radio technology designed to address the Low Power Wide Area Network (LPWAN) IoT market
LPWAN	Low Power Wide Area Network
M2M	Machine-to-Machine
NAT	Network Address Translation
P2P	Peer-to-Peer
PDN	Packet Data Network

PKI	Public Key Infrastructure
QoS	Quality of Service
Sigfox	A Low Power Wide Area Network (LPWAN) technology and ecosystem designed to address the LPWAN IoT market
SIP	Session Initiation Protocol
SMS	Short Message Service
SMSC	Short Message Service Center
SSDP	Simple Service Discovery Protocol
TAL	Threat Agent Library
TLS	Transport Layer Security
UPnP	Universal Plug and Play
VLAN	Virtual Local Area Network
VoIP	Voice over Internet Protocol
VPN	Virtual Private Network
WAN	Wide Area Network
WEP	Wired Equivalent Privacy
WPA2	Wi-Fi Protected Access 2
Zigbee	An IEEE 802.15.4-based specification for a suite of high-level communication protocols used to create personal area networks with small, low-power digital radios