CHAPTER

SAFETY REQUIREMENTS IN THE INTERNET OF THINGS

6

This chapter discusses safety risk requirements in the IoT and how they are related to security requirements.

Safety is a distinct requirement/characteristic in the IoT, having to do with areas not typically addressed by IT, but critical in the IoT given the cyber physical, logical kinetic interconnections. Safety is more about physical-device (vs software-system) resilience, and predictability of performance and failure (logically and physically).

Safety is intertwined with security and many of the other chapters will discuss requirements that have as much to do with safety as it does with security. For instance, safety is absolutely related to matters such as availability of confidentiality, but also the industrial design and usage context of an IoT system or service.

Safety will also be as much a part of the risk management associated with matters like supply chains, provenance, and the complete life cycle of a product or system, whether the product or system is to be used in the workplace, in the household environment, or for recreational activities.

Safety in the IoT might be seen more prosaically as something that is a physical outcome of a logical or *cyber* event. The following sorts of events might be considered a safety impact for humans:

- Explosion and burning.
- Allergic reactions, for instance to wearable things or to environmental conditions that change as a result of things like climate control systems.
- Sensory impacts (degrade or harm hearing, or especially sight), such as augmented reality systems and services that temporarily or even permanently degrade senses because they are too intense or close.
- Infections and tumors, again related to wearable or implantable devices in the IoT used for therapeutic applications.

> The goal of this chapter is to highlight elements that are associated with security but are more typically discussed in a safety context. Put another way, this chapter is intended to introduce IoT risk managers to requirements that may be unfamiliar to those coming from an enterprise IT security environment, versus an industrial environment.

Finally, the IoT safety requirements defined in this chapter will take on different complexions depending on the intended use case of the IoT service. For instance, different types of end users will have greater or lesser understanding of safety and may require different design assumptions around the need for automation of safety functions. Similarly, different operational environments,

RIoT Control. DOI: http://dx.doi.org/10.1016/B978-0-12-419971-2.00006-6

such as home, office, factory, or rugged-outdoor systems, will require more or fewer layers of supplementary safety and security systems, depending on the use case. This indicates that broad-based assumptions about safety and security will not work in the IoT, because in some cases a failure in one security or safety system may be compensated for in other adjacent systems, and in other cases it may not.[1]

SAFETY IS NOT EXACTLY THE SAME AS SECURITY

Ask any industrial control system (ICS) engineer whether enterprise IT security standards and processes are useful in their environment, and he/she is likely to say "partially but definitely not completely." ICS security practitioners have for many years rejected the overtures of *IT security* experts and standards, claiming that ICS is not the same and has different requirements.

They were right. They are right! The lessons learned from those early encounters between ICS and IT now extend to the IoT—which has *combined* the two practices inextricably:

$$ICS + IT = IoT$$

To try and summarize it: ICS and IT have different performance and reliability requirements. ICS especially uses operating systems and applications that may be considered unconventional to typical IT support personnel. Furthermore, the goals of safety and efficiency can sometimes conflict with security in the design and operation of control systems (for example, requiring password authentication and authorization should not hamper or interfere with emergency actions for ICS).

In a typical IT system, data confidentiality and integrity are typically the primary concerns. For an ICS, human or property safety and fault tolerance to prevent loss of life or endangerment of public health or confidence, regulatory compliance, loss of equipment, loss of intellectual property, or lost or damaged products are the primary concerns. The personnel responsible for operating, securing, and maintaining ICS must understand the important link between safety and security.

In a typical IT system, there is limited or even no physical interaction with the environment. ICS can have very complex interactions with physical processes and consequences in the ICS domain that can manifest in physical events.

Safety as an IoT requirement also addresses one key aspect of system behavior: protection against entropic (random) faults of an unintentional nature.

The following safety requirements might overlap and be interdependent with other requirements to follow in this book, but they are worth understanding independently because of the critical nature of safety in the IoT.

PERFORMANCE

Information technology (IT) is full of false claims about performance, which will represent a large safety risk to the IoT. Vendors of IT hardware and software alike will publish claims about

[1] ISO EIC Guide 51—*Safety Aspect—Guidelines for Their Inclusion in Standards*, third ed.

performance metrics that simply cannot be replicated. This is all too common; however, industry has learned to adapt to this chronic overstatement of performance by discounting vendor claims, requiring (expensive) trials and proof-of-concept demonstrations, and generally over provisioning infrastructure.

Customers often buy a network device expecting that it will perform at 1 Gbps, for instance, only to find that once they configure it the way they need its performance drops to half or even less! Similarly, organizations invest in software expecting that it will handle (again, just an example) 100 transactions per millisecond, only to find that the vendor performance claims are supported only with very specific hardware configurations that are not appropriate to the customer environment.

In the IoT, where the logical-kinetic/cyber physical interfaces predominate, performance will be about features and metrics like: time criticality, delay, or jitter—reliability of performance; whereas some of the IT-related metrics like maximum throughput might not be important. We will discuss such performance metrics subsequently in this section.

In the IoT, performance of endpoint, gateway, network, and cloud/data-center elements needs to be as advertised by product and service vendors.

Clarity of performance in products and services is an essential requirement of the IoT. When it comes to performance in the IoT, both product and service vendors need to be aware that fudging the numbers or being deliberately vague or deceptive drives untold risks.

RELIABILITY AND CONSISTENCY

ICS includes safety instrumented systems (SIS), which are hardened information elements built for high reliability and associated with failing safely and predictably. This is what the IoT needs.

Conversely, IT elements from the enterprise network and data center (DC) environment are typically not built for high reliability; they are integrated into *high-availability* (HA) pairs and clusters. HA is a cheap substitute for hardware and software reliability because it is assumed that even with poor reliability, most (or at least half) the elements will remain functional after a failure in one element.

IT design conventions related to high availability and clustering do not extend well into the more remote parts of the IoT, such as gateways and endpoints, where the economics (business cases) just do not make sense and the services cannot be deployed based on safety techniques that rely on doubling up on infrastructure.

Many ICS processes are continuous in nature and must therefore be reliable. Unexpected outages of systems that control industrial processes are not acceptable. ICS outages often must be planned and scheduled days or weeks in advance. Exhaustive pre-deployment testing is essential to ensure reliability of the ICS.

In addition to unexpected outages, many control systems cannot be easily stopped and started without affecting production and safety. In some cases, the products being produced or equipment being used is more important than the information being relayed. Therefore, use of typical IT strategies, such as rebooting a component, are usually not acceptable solutions due to the adverse impact on the requirements for high availability, reliability, and maintainability of the ICS.

Similar to the requirements for performance, reliability in the IoT needs to come with more significant and robust specifications with regard to reliability. Measures like *mean time to replacement* (MTTR) or *mean time to failure* (MTTF), which are common in the network and DC world, will need to be extended out towards the edges of the network, in which devices cannot be deployed in HA or clustering designs.

Overall, safety in the IoT will require that gateway elements especially, but also endpoints, become more reliable and consistent in stand-alone performance.

NONTOXIC AND BIOCOMPATIBLE

Much like concerns today about batteries, compact florescent lights, mercury thermostats, and ozone-depleting air conditioning units, a substantial safety risk in the IoT will be associated with the impact of materials used to build IoT devices.

The IoT in many cases will be about devices that are destined to be absorbed into the environment or embedded into living tissues and bodies. For instance, environmental sensors might be deployed with the expectations and business assumptions that once they cease working, they will be left in place to simply decay and disappear. Alternately, the current generation of *wearable* technologies will inevitably evolve into other devices that will be placed more directly on the skin for longer periods of time or will be embedded. The implants of today will certainly become connected, for the purposes of better monitoring, diagnostics, and management.

IoT devices will need to be designed with environmental safety in mind. Devices made of toxic materials will probably engender rougher regulation and monitoring of their distribution, use, and disposal—raising costs.

Safety of the IoT will have a lot to do with not only how the devices act and respond to commands, but what they do to the environment in which they operate, both during and after their useful life.

The need to start engineering devices with newer, specially developed biocompatible materials will potentially mean that other safety features such as reliability and predictability may suffer because the world of information processing and computing is very demanding in terms of physical stresses. Moves toward more environmentally friendly, safe materials in the construction of IoT endpoints will absolutely have effects on the data processing and management assurance of those devices, if for no other reason than that it will reflect a change in the system.

Understanding the safety and risk trade-offs associated with the use and adoption of new, safe materials in the IoT will be critical for risk managers.

DISPOSABILITY

Related to the issue of toxicity in IoT safety is the matter of safety and disposability. What happens when the device reaches end-of-life, is made obsolete, no longer wanted, or is defective and cannot be repaired? From a safety perspective, the environmental issues are clear—but the linkages between safety and information security associated with disposability may not be apparent at first glance.

In the security world, hardware and software disposal is a well understood security process and requirement. Device, system, and service owners in the IoT all must be sure that information is destroyed in the process of disposal of IoT devices, and unauthorized access is not granted to personal or proprietary information (operating systems, configurations, designs, and so on). Many spectacular information security breaches have occurred due to poor or missing disposal practices.

There are disposal issues around safety, too, that will have cascading impacts to IoT security and risk management overall.

Disposability will affect safety in the IoT in relation to elements like the biological and environmental toxicity of the IoT endpoint and edge devices. Will they poison the users? Will they become hazardous once they reach landfills or incinerators in the thousands or millions, or once they get decommissioned but left in place, whether embedded into asphalt or embedded into living flesh?

For instance, in the case of wearables or devices that might get embedded into objects or people, there will evolve clear requirements for mechanically and environmentally stable materials such as:

- Batteries and energy collection and conversion parts
- Conductors/wires
- Processors and memory
- Insulators
- Packaging, housing, and monitoring and control interfaces
- Substrates and functional materials

While safety may dictate that certain materials be used and others be avoided, the impact on information security may be hard to balance as a requirement. For instance, tamper proofing or tamper resistance of information processing or storage parts may require materials that do not meet safety and disposability criteria! Or, disposable battery types may not support the availability requirements and service levels of information security.

SAFETY AND CHANGE MANAGEMENT IN THE IoT

Change management is paramount to maintaining the security of both IT and IoT systems, and is also applicable to both hardware and firmware. As every student of information security knows, patch management required to fix vulnerabilities and other security-impacting flaws is a major supplicant to change-management processes.

Unpatched systems represent one of the greatest vulnerabilities to an IT system. Software updates on IT systems, including security patches, are typically applied in a timely fashion based on security policy and procedures intended to satisfy compliance (organizational) requirements. These procedures are often automated in enterprise IT, using server-based tools and auto-update processes.

Yet, software updates in the IoT cannot always be implemented on an automated basis. In the IoT, each software update may have safety-critical dependencies associated with it, whether it be associated with downtime for patching or the fundamental stability and performance of the IoT system after patching. IoT updates will need to be thoroughly tested and sanctioned by the potentially multiple stakeholders, such as the various equipment, application, and service vendors, as well as the user of the application.

The IoT system, as a whole, may also require revalidation and certification as part of the service level agreement (SLA) and compliance processes stipulated in contracts to high-assurance clients like governments or banks.

Change management process from IT might be the basis for change management in the IoT, but wholesale adoption would be inappropriate and such practices would represent risk to an IoT system or service.

DIVISIBILITY OF SAFETY AND SERVICE DELIVERY UPDATES AND LONGEVITY

There are conflicting requirements between safety and many of the other risk management requirements discussed in this book! Where possible, an upgrade path and methodology for safety functions versus service functions should be separate.

IoT system or device owners should be able to update or upgrade service-related software without impacting safety-related software. Where both capabilities require upgrade or patching, these processes should be divisible.

Not only should safety versus service patches be divisible, but safety patches (unless related to a critical flaw) should be forward compatible with service patches indefinitely. For instance, a safety patch should never be mandatory unless it is related to a critical flavor vulnerability related to performance, reliability, or efficiency of safety systems.

It is possible that evolving technology and design flaws in both hardware and especially software impact information security and risk management requirements, without impacting the safety systems and requirements. The safety system, due to simplicity and clarity of requirements and purpose, may continue to work just fine in the face of a failed service-delivery platform!

Therefore, IoT safety can be more difficult to manage if it is inextricably linked to IoT security management, such as updates. Here are a couple of examples where information security upgrades to the system might be delayed due to safety-critical services and functions:

- Old systems that can’t be patched or upgraded without risk to the safety systems
- Old or expensive IoT systems might not have the luxury of development and test environments, so patching in operational system means risking not only the service and production processes, but associated safety processes too. (For instance, a patch works and the service platform is upgraded but the safety processes become unreliable!)

STARTUP AND SHUTDOWN EFFICIENCY (MINIMIZATION OF COMPLEXITY)

From a safety perspective in the IoT, the ability to start and stop a process quickly will be a major risk management requirement. This in turn will implicate not only the design of the IoT endpoints themselves, but the gateways, networks, and cloud services that support them.

The ability to start quickly will be important for IoT devices involved in kinetic motion and movement control. For example, where they may not be active until a person or object comes into range for the purposes of saving power, or where they only become active after sensing an abnormal physical condition like an imminent collision or change in the physical environment. However, once the device is called into action it must be available very quickly—potentially meaning that it must move from a dormant state to an active state in milliseconds.

Under these conditions, an IoT device might be required to forgo security controls that would otherwise be built into the device, gateway, or network communications. Encryption technology on the device is a good example. If the time and energy it takes to decrypt an instruction set on a small, constrained IoT device will take the startup time from 2 to 4 ms, it is possible that service level requirements will be missed!

Very much related to this matter of startup will be the gateways that might support bootstrapping of IoT devices at startup or installation, and the networks and cloud systems that will provide provisioning information and configuration details. If these systems cannot support the startup safety requirements in terms of performance, alternatives will need to be found!

The ability to shut down quickly will similarly have significant safety effects in the IoT and will be a function of not just the endpoint devices, but also the gateways, networks, and clouds that they rely upon for instructions and connectivity.

Shutting down quickly is not so much a matter of the performance of the IoT, but about *failing safely*, and reporting a shutdown in a timely manner to centralized management tools. As another example, when a device is deemed to have breached its stipulated performance parameters, it needs to shut down quickly and allow a redundant system to assume the service functions. Or, in the event that there is no redundant system, it needs to fail safely. (See the next section.)

Consider a health monitoring system. It will probably need to be capable of monitoring a patient’s pulse rate or blood pressure very accurately. If the IoT system or service starts to see anomalous readings indicative of an impending device failure from one set of sensing devices connected to the patient, those devices will need quickly to shut down so not to contaminate the pool of good data already collected by the systems in place.

Alternately, in an ICS managing thousands of tons of molten metal, or a transportation system directing thousands of vehicles moment to moment, once a defective device is detected it needs to be removed from the system as quickly as possible.

Failing and defective devices must be stopped quickly and fail safely for one final reason—the complexity of the IoT and the interconnected systems means that bad data and inappropriate operations by even the smallest of devices can have untold effects. Chaotic effects.

As with startup requirements, shutdown and failover requirements may supersede information security requirements for things like encryption, platform validations, device authentication or log-off, clean session termination, and so on. In turn, the lack of these information security controls can make a device more prone to a variety of attacks such as man-in-the middle or masquerade.

Startup and shutdown safety and its relationship to IoT information security is yet another safety balance to be sought by risk managers in the IoT.

FAILING SAFELY

Failing safely means that a system element will stop working in a predictable manner: a manner that has been accounted for in the service design and will result in a physically or logically safe state, post-failure. In other words, physical damage or system disruption is minimized, controllable, and foreseeable.

For most IT devices on the Internet, we really have no idea how they will fail. The hardware platform manufacturers and the software typically do not collaborate to anticipate a service-to-device failure, let alone report it to users.

Even if there was a will to determine how a given IoT device or system might be developed to fail safely, it would be impossibly complicated and expensive given the range of software vendors and hardware vendors for things like servers in the DC, network elements (like routers, firewalls, domain name servers), gateways, and endpoints. As a result, when IoT devices fail, it is very likely that their state will be unpredictable and therefore not predictably safe!

Frequently, devices will freeze and lose data and cutoff connectivity. Sometimes, they will be set to *fail open*, meaning, that is, they stop working, and then any information management operations (like security) they performed cease—but data will continue to flow. In other words, they are designed to continue to pass untreated or insecure data rather than stop the data flow altogether. Sometimes, this makes lots of sense. Sometimes, it does not.

A possible outcome is that some IoT devices will continue to function, but will become *zombie* devices that are no longer responding to external commands from owners and administrators—they just keep doing whatever their last instruction told them to do. In a way, this is worse than failing; this is a runaway train scenario!

In the industrial control world, *failing safely* means something. It means that if a device is no longer behaving the way it is expected to behave, it can be shut down into a predetermined state. That state may be open, closed, or perhaps limp home, where it curtails all functions except the most basic in order to minimize collateral impacts until help arrives.

More attention needs to be paid to failing safely in the IoT and what this means from an information security perspective, especially on endpoints and gateway devices, where security will be first applied.

ISOLATION OF SAFETY AND CONTROL FROM SERVICE DELIVERY

To the extent possible, maintaining isolation of IoT safety systems from operational process management systems is the way to go. This is long-standing best practice in the industrial control world. The difficulty with this requirement for the IoT is the competitive drive for low cost, and the related need to remain efficient in operations. Requiring endpoint devices or gateways to have logically or physically distinct safe versus administrative interfaces can add substantially to costs, both capital and operational.

Safety systems often use the same technology platform as the IoT service-delivery systems, meaning that IoT service vulnerabilities may well be common-mode failures to safety systems, allowing an attacker to compromise both service delivery and control, and safety logic at once or using the same tradecraft.

For instance, an existing issue is that engineering workstations are used to configure both IoT (and industrial control devices) and safety systems—which means that a threat agent could compromise the IoT assets and the safety systems by gaining access to a single workstation. This issue is amplified by the prevalence of *commodity* operating systems, which may have potentially thousands of known vulnerabilities that require only modest amounts of skill of which to take advantage.

In order for the IoT safety systems to function properly, they must also be connected in some way to the IoT service-delivery functions in order to monitor performance, and determine whether safety logic must be invoked. As such, there really is no such thing as a disconnected safety system for the IoT.

From a requirements perspective, it is important for system designers, engineers, and managers to understand that combining safe and service-delivery functions without a thought to any form of isolation creates significant risks. Here is a scenario: knowledgeable attackers could bypass or suspend safety logic without touching service-delivery functions in anyway: business as usual. At that point, they merely wait for a *normal accident* to occur on a random schedule and allow the situation to unfold as it would without safety systems in place.

SAFETY MONITORING VERSUS MANAGEMENT AND SERVICE DELIVERY

Safety systems in the IoT are generally designed with a single purpose in mind: avoiding dangerous situations in the environment (logical-kinetic/cyber physical) by stopping or shutting down services and processes if unsafe conditions develop. Additionally, safety systems are typically implemented as compensating controls for known or anticipated hardware or software failure rates. These failure rates are established through recognized and generally accepted good engineering practices adopted by both asset owners and vendors, driven by industry standard such as ISA-84, IEC 61508, IEC 61511, and others.

In this regard, safety monitoring functions and capabilities in endpoints, gateways, networks, and cloud services in the IoT will be developed for watching safety parameters, not managing or administering safety systems. Actually, management and administration of safety services, functions, or systems in the IoT will be strictly limited.

Therefore the requirement for safety and monitoring is that as far as practical and possible, segregation and access controls the software and hardware used for safety versus service-delivery management is required.

The opposite condition would be to allow any stakeholder who has access to a service-delivery function to have access to safety functions at the same time through the same interface. For example, there is one interface used to for both safety monitoring and service-delivery management.

This single-interface design could be typical for a low-cost IoT device; however, the problem comes in when this single monolithic interface fails and takes both safety monitoring and service management with it at the same time—meaning that control is not only lost, but safety-critical visibility and awareness is gone in the same instant.

RECOVERY AND PROVISIONING AT THE EDGE

In a typical IT system, the primary focus of security (and especially recovery) is protecting the operation of centralized IT assets, such as the assets in the DC or the cloud: databases, file systems, servers, and the like. In the IoT, this condition may be reversed for safety reasons. In the name of safety, an edge device may take on more priority in the recovery process and procedures relative to the centralized assets.

In many IT-centric, conventional Internet architectures, information stored and processed in the DC or cloud is more critical and is afforded much more protection than information stored and managed at the edges. For IoT systems and services, edge devices (such as gateways) need carefully to be considered in business continuity prioritizations because they may be directly responsible for controlling the safety-critical functions and services to endpoints. The protection of the central services is still very important for safety because the central server itself might contain critical instructions for safety management.

Recovery of gateways may involve a number of processes, which amounts to re-provisioning a device in the field either physically or logically. While provisioning will be a critical factor in the development of IoT services, the service levels associated with provisioning may only be factored in within the context of new devices coming into the system. In the IoT, a safety requirement might exist in a given service or system that associated specific service levels for recovery and re-provisioning of gateways. (Re-)provisioning may be as much about safety and merely turning up the service for the first time.

MISUSE AND UNINTENDED APPLICATIONS

In Chapter 7, Confidentiality and Integrity and Privacy Requirements in the IoT, we will discuss documentation and reporting as a security requirement in the IoT. But it is also a safety requirement related to misuse that must be considered at the same time as the security requirement, because such misuse can have dramatic safety implications for the IoT and cascade from information security impacts to physical safety and security impacts.

Without fail, it will come to pass that IoT devices, systems, and services will be used for unintended purposes, in the manners unimagined by designers and service providers. This is known human behavior and should be considered in the design process for safety as well as for security. People will misuse IoT services for all manner of reasons, from mischief to fraud to negligence to laziness to ignorance, due to handicaps, misunderstandings, environmental conditions (see Chapter 10, Usage Context and Environmental Requirements in the IoT, on context),or simple errors and omissions.

The known habit of people applying weak or no passwords to protect system configuration has to be accounted for in safety discussions. While it might be improbable or even inexplicable that an IoT device might be configured by the users in a way to harm the users—it should be expected, and safety documentation and warnings incorporated carefully.

For instance, home thermostats are rapidly coming onto the IoT because of the potential savings associated with energy management. Yet most of these thermostats have rather weak safety features when it comes to malicious or accidental misconfiguration. The furnace is accidentally turned off in the middle of a Canadian winter while the family leaves on vacation; resulting in massive structural damage to the home due to frozen and exploded water systems. Or the heat settings are adjusted by a user thinking he/she was using a Fahrenheit scale when Celsius was applied, resulting in a hugely high setting that forces the furnace to work to the point of failure (and huge gas bills).

Good usage instructions and technical safeguards will be a requirement for, and complement to, information security controls in the IoT.

SUMMARY AND CONCLUSIONS

Safety in the IoT is related to IT security in that it is concerned with intended use and reasonably foreseeable misuse, failure, and malfunction.

Where safety differs from security is in key areas such as the need for:

- Availability over confidentiality in performance.
- Being reliable and consistent in performance under normal conditions as well as failure conditions.
- Management of toxicity and disposability; this will heavily influence IoT device design in the future, potentially exposing the IoT to security threats such as tampering, tapping, or service levels.
- Management of change management conventions in IT; these do not translate well into the IoT or process control worlds.
- Starting and stopping service levels introduced as requirements in a way not typically seen in IT systems.
- Failing in a manner that does not jeopardize safety requires much more engineering—where failing has ad hoc outcomes in many IT systems.
- IoT safety and monitoring systems that should be isolated or at least developed in stovepipes (as far as practical) from IoT operational and management systems—including at the endpoint.

Safety in the IoT is as much related to consumer protection and risk management as security. It is part and parcel of determining the risk posed by consumer and business products and services. Consideration should be given for IoT products and services that are intended for, or are used by, vulnerable consumers who are often unable to understand the hazard or the associated risk. But at the same time, in making devices safer (from a safety perspective), they potentially become less secure.

On the other hand, the interests of the service providers and device makers may lie in the area of security due to business risks, but a balance with safety must be addressed and risk managers must consider both safety and security at once, as a whole.