Connectivity, Cybersecurity and Medical Devices: What are the Threats?

Developers of medical devices are taking inspiration from the use of connected technology in consumer products; realising that smarter devices offer huge benefits to patients, POC workers, and manufacturers. Today's medical devices can be connected within wider network infrastructure, perform remote upgrades to software with relative ease, and be accessed through smartphones.

Whilst these technologies can be used to increase the functionality of devices – for example, improving device usability, or allowing manufacturers to perform remote post-market surveillance – they can also open them up to cyber-threats.

We will look at ways to identify common risks and vulnerabilities in medical devices and how to defend against them, to help put you on the right development path.

An Overview of Cybersecurity Risks

As medical device manufacturers, designers, and distributors, many of us are familiar with the concepts and processes of risk management throughout the lifecycle of a medical device, as described in ISO 14971. During medical device development, risks are identified and subsequent mitigations are devised to reduce the probability, or severity, of these risks. Following this, manufacturers will then monitor the device throughout its lifecycle, to assess new threats and the effectiveness of any mitigations.

Cybersecurity risks are handled in a similar way. In the United States, the National Institute of Standards and Technology (NIST) provide a cybersecurity framework[1] which, in some ways, is similar to our understanding of ISO 14971.

At a high level, the cybersecurity framework involves five over-arching activities. Identify threats, assets, and impacts; protect assets against these threats using appropriate safeguards; detect cybersecurity events when they occur, and devise ways to detect events; respond to, and manage, cybersecurity events; recover from events, minimising their impact.

Whilst not specific to medical technologies, the cybersecurity framework is targeted towards cybersecurity risks in critical infrastructure, in which we can include medical devices. It recognises that the impacts of security incidents can be wide-reaching and highly variable.

The risks relating to a device depend largely on its purpose and the specific design employed, however it is still possible to generalise some of the possible harms associated with cybersecurity across all medical devices.

Direct Patient Harm

In the worst cases of cybersecurity incidents, an attacker could intentionally misuse your device in an attempt to cause harm to patients, caregivers, and others involved. For example, if a wireless interface is used to control your device functionality and trigger therapeutic behaviour, an attacker could use this interface to control and manipulate the device. Without protection, the interface is directly vulnerable to attack.

Loss or Manipulation of Data

Where sensitive or personal data is in transit, it can be exploited by attackers to steal identities, extort individuals, or sell to unscrupulous organisations. In many cases we already protect this data within our IT systems, as it falls under the scope of legislation such as GDPR.

Attackers may also manipulate critical data; medical data could be modified by attackers in a way which leads to changes in the patient's treatment. In current times, consider the manipulation of COVID-19 test results. A false-negative test leads to potentially infectious patients continuing in their lives, with well-known consequences.

Denial of Service

A "denial of service" (DOS) attack is one which is used to block your services. The most common takes the form of server downtime due to an enormous number of access requests in a short period. Alternatively, as we saw with the WannaCry attacks on health services, services can be blocked by ransomware which holds your system hostage.

The WannaCry attacks on healthcare providers showed the wide-reaching effects when critical services are inaccessible, with delays to surgeries and treatments for thousands of patients worldwide.

Leapfrog Attacks

In connected systems, no point in a network stands in isolation. This allows the connected system to distribute information between nodes and improve patient outcomes. Unfortunately, for attackers it also provides a mode of entry for a much wider system. By attacking one poorly protected device, it may be possible to attack the wider system. A system is only as secure as the least secure device within it. It is therefore imperative that all developers strive to make their devices as secure as possible.

Loss of Intellectual Property

It is possible for your software to be read and interpreted directly from the device, allowing attackers to understand how your device works. In doing so, attackers may re-create your functionality; thus taking advantage of the development work your team has completed in order to reach the market in a significantly shorter period.

Identifying and Assessing Vulnerabilities

While there is a wide array of cybersecurity risks in medical devices, it is important to remember that each individual device, system, and product line will introduce and observe unique risks.

When assessing risks associated with cybersecurity, it is important to assume that an attack *will* occur and that any vulnerability *will* be abused. It can be useful to consider these risks under the assumption that no mitigations are in place. Where mitigations exist, we can assess the likelihood of them failing; for example, data encryption can fail if a weak encryption method is used.

The concept of "guaranteed" risks is applied commonly during medical device software development, with a similar rationale. This means ensuring that all cybersecurity risks are taken seriously and that mitigations are applied to prevent (as far as possible) these risks being exploited. Such a rationale helps to account for the nature of some cyber attackers, who may attempt to break into a device "for fun".

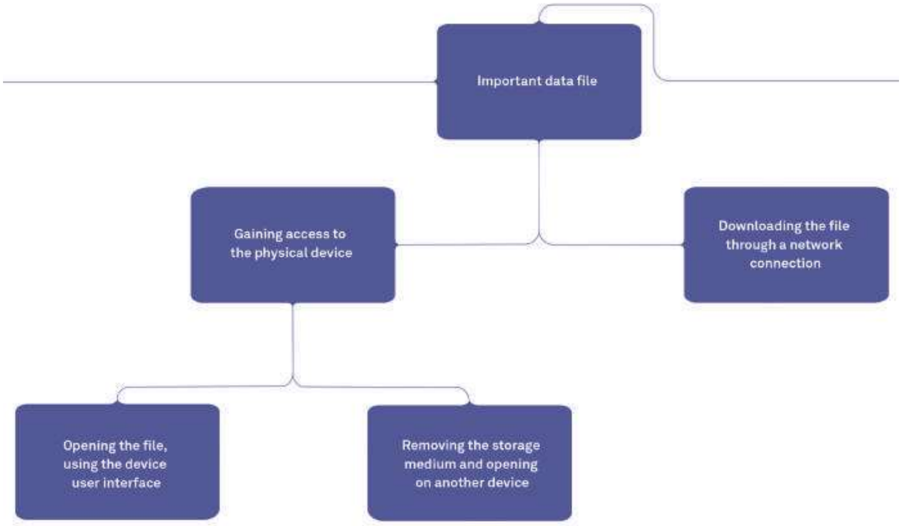
Similarly to device safety risks, we can examine risks from two directions; the top down, and the bottom up.

Top-down

Top-down cybersecurity assessments look at the overview of the system, identifying higher-level features or resources which could be exploited, and then identifying how these could be accessed or manipulated. To start this, try asking yourself what could happen if an asset was stolen, corrupted, or misused.

Example assets include patient information being used to blackmail the patient. High-power energy sources, such as X-Ray and RF, could be used to cause direct harm or disruption to nearby equipment. Clinical data such as test results or images, when manipulated or corrupted, could lead to an incorrect treatment regimen being applied.

Having understood our assets, we look to understand the routes of attack. A method of documenting this is to record an attack tree; as described by B. Scheneir[2]. Attack trees examine the different ways that an asset can be attacked or that an event can occur. This breakdown continues until we have exhausted the possibilities. Consider a sensitive data file on a device which currently has no protective measures. An attack tree diagram clearly illustrates how the file data can be stolen.

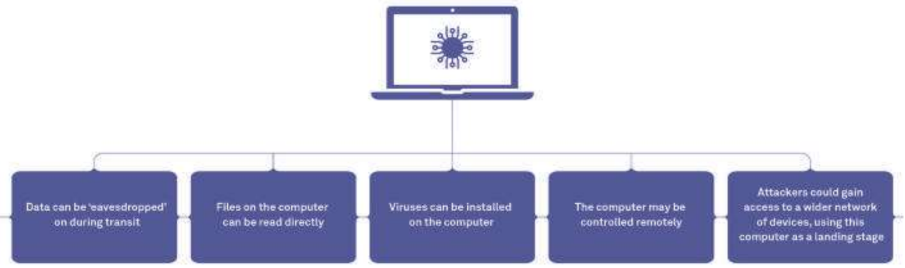

Important data file
Gaining access to the physical device
Downloading the file through a network connection
Opening the file, using the device user interface
Removing the storage medium and opening on another device

The top-down approach provides a robust way to identify potential weaknesses in the early stages of projects, focusing from a higher level and ensuring that we consider cyber-risks when identifying device requirements. It should be repeated regularly during device development, to ensure new assets are identified and managed throughout the lifecycle.

Bottom-up

The bottom-up approach is commonly performed during the detailed design, and implementation, phases of the development cycle. The assessment begins from a finer level of detail than the top-down approach in order to identify harms caused, or assets accessed, when failure occurs. This can be considered similarly to a failure mode and effects analysis (FMEA), where teams examine the risk of failures of individual components from a design which is already well understood. Our mitigations focus on ways to prevent, or detect, failures of specific design elements.

If we take a computer as an example, we examine the details of our system to identify how it could be misused or attacked. We can identify that it has a network connection available for use, so the next step involves looking at how it could be exploited:


Data can be 'eavesdropped' on during transit
Files on the computer can be read directly
Viruses can be installed on the computer
The computer may be controlled remotely
Attackers could gain access to a wider network of devices, using this computer as a landing stage

This bottom-up approach can help you to identify further risks which a top-down approach may miss, as you are examining the methods and interfaces which an attacker can use directly. By using both approaches, the goal is to discover as many security vulnerabilities as possible, and to then determine the mitigations for them.

Common Vulnerabilities and Mitigations

It is rare to encounter an entirely new cybersecurity problem during design; this is where the common vulnerabilities list becomes useful. This highlights well understood vulnerabilities, such as specific network protocol exploits, which can be used by teams as a reference when identifying potential risks. The UL2900 cybersecurity standard expects that all common vulnerabilities which apply to your system have been mitigated against, so this is an important starting point.

So how can we mitigate against the risks we have identified? Whilst each application will have specific needs, we can consider some common concepts which should be considered during device development.

Data at Rest

Many devices store data at rest which could be considered sensitive, for example user passwords, clinical data, or platform data. Attackers could access such data directly through a device's hardware, or the user interface. The most common mitigation to protecting data is to encrypt the data, using robust encryption mechanisms such as AES256.

Encrypting data is a key way of reducing opportunities for attackers to understand it. This involves scrambling the data so that only a device with an encryption key can interpret it. It is important that encryption keys are, themselves, protected from attackers – you don't leave the key to your house under a plant pot! Whilst in some cases, storing encryption keys in protected memory may be acceptable, in many cases a trusted platform module (TPM) may be required to ensure that keys are protected adequately.

TPMs are bespoke hardware modules which provide a wide range of security functions, including managing cryptographic key storage and exchange, encryption and decryption of data, and the authentication of users and data. These can be integrated within your device to ensure high levels of device security. They are also produced by a wide range of manufacturers, allowing development teams to select the most appropriate device for their application's needs.

Data in Transit

Data in transit, meaning data transferred between devices, is vulnerable to eavesdropping, where an attacker can steal information while it is being sent. A simple solution, as discussed for data at rest, is to ensure that data in transit is encrypted from "end-to-end". The principle of end-to-end encryption is simple; whenever data is being transferred between parties it is encrypted.

In a system which is encrypted end-to-end, the encryption keys are only known by parties which need to use the data. Consider data transfer between two computers on a network which must be secure. The sender PC can encrypt the data before transfer using a key known only by the sender and recipient. This data is handled by the server but cannot be read as it does not understand the encryption keys.

Until now we have considered a system where the encryption key is shared by both the sender and recipient of data. In reality within this system (known as symmetric key encryption) the shared encryption key may act as a single point of failure for multiple devices. A more secure approach is to use asymmetric key encryption which, in brief, allows two devices to determine an encryption key which is unique to each communications session, without needing to know the encryption key ahead of time. An example of asymmetric key exchange is the Diffie-Hellman protocol. Where feasible, developers should aim to use these asymmetric key exchange protocols for inter-device communications.

Data Authenticity

Alongside data being observed, another option for attackers is to disrupt data in transit and to modify it. This can be performed through "man in the middle" type attacks, or through random data corruption from electromagnetic interference. Corrupted data can introduce significant risks, for example incorrect test records, or incorrect instructions to remote modules. We can identify corrupted data by attaching robust signatures (such as SHA256) to the data. With these, the recipient confirms that the data is valid by calculating the signature of the received data – if this does not match the provided signature then the data cannot be authenticated.

We have already examined the idea of encrypting our data during transit, but how do we know that the system providing our new software image is from a genuine source? This is the second part of authentication; determining if the data is valid and from a valid source.

Here we can take inspiration from website and server certificates, used to confirm that the communication is with the correct host. Once an encrypted communication session has been established, we can exchange authentication data between devices, establishing that we are communicating with a valid recipient.

User Access

Most of the examples we have discussed have been around remote attackers gaining access to data through networked interfaces, but in reality the simplest way to access data on a device is to gain access to the device itself. In most computer systems, we are familiar with the use of passwords to secure a device; however, these can become a point of entry for attackers if they are discovered.

The greatest protection here is to ensure that strong password policies are used on devices. Each device and device user should have a unique password, and access to data should be restricted where appropriate.

The use of system-wide default passwords for all devices is highly discouraged, as it offers attackers an opportunity to access all devices within an ecosystem. Devices will need some form of default password when configured, but this should always be a unique password to the given device. In California this is a legal requirement[3] for all internet connected devices, and should ideally be factored within all manufacturing and distribution processes.

In Summary

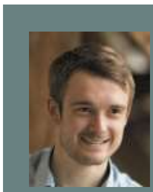
In the modern, connected world of medical devices, ensuring that the device remains secure must be considered part of the overall risk management process. We have discussed some of the risks associated with cyber-attacks, common vulnerabilities which can be exploited, and examples of mitigations which can be applied.

Unfortunately, even if we protect against all foreseeable methods of attacks, cybercriminals will find novel ways to break through your defences. The key to ensuring a secure device lies in starting the risk management process early, planning how security threats can be removed once they are understood, and continuously monitoring and evaluating the potential for new threats.

By recognising assets and mechanisms for attack in the early stages of design, your system architecture can include cybersecurity from the outset.

REFERENCES

1. https://www.nist.gov/cyberframework/online-learning/five-functions
2. https://www.schneier.com/academic/archives/1999/12/attack_trees.html
3. https://techcrunch.com/2018/10/05/california-passes-law-that-bans-default-passwords-in-connected-devices/



Thomas Watts

Thomas is an electronics and software engineer at Team Consulting, where he specialises in embedded software development for medical devices. Before joining Team, Thomas worked for SLE in London; a company focusing on neonatal ventilator systems. He has an MEng in Biomedical Engineering from Imperial College London.