Independent Submission
Internet Draft
Intended status: Informational
Expires: May 2, 2021

M McFadden
internet policy advisors ltd

November 2, 2020

Endpoint Security Classification
draft-mcfadden-endpoint-classification-00

Status of this Memo

Copyright Notice

Abstract

It seems reasonable to suggest that, despite the huge variety of types of endpoints on the Internet, there are categories of similarity. These categories are important because categories of endpoint devices may share particular advantages or limitations for endpoint security. This draft attempts to suggest a classification of endpoints as a foundation for further work on operational security. The goal is to identify classes of endpoints with similar characteristics. Those characteristics may lead to the discovery that the devices in a particular category share similar characteristics for endpoint security.

Table of Contents

1. Introduction

A document entitled "BCP 72 - A Problem Statement [I-D. draft-mcfadden-smart-threat-changes-01] suggests that the Internet's threat landscape has changed significantly since the publication of BCP 72. One of those changes is the evolution of security at endpoints. From an operational viewpoint, the end-to-end principle has previously focused activity on endpoint security.

Operational experience has identified limitations of endpoint-only security solutions. Significant changes in technology, economics and protocol development have impacted the provision of endpoint security.

There are an enormous variety of endpoints on the Internet. It seems a daunting task to try to make generalizations about endpoint security when there is such diversity in the types of devices connected to the Internet.

However, it seems reasonable to suggest that, despite the huge variety of types of endpoints, there are categories of similarity. These categories are important because categories of endpoint devices may share particular advantages or limitations for endpoint security.

This draft attempts to suggest a classification of endpoints as a foundation for further work on operational security. The goal is to identify classes of endpoints with similar characteristics. Those characteristics may lead to the discovery that the devices in a particular category share similar characteristics for endpoint security. While a general-purpose taxonomy of Internet endpoints might be useful in a variety of settings, it is not the intended goal of this document.

In addition, this document does not attempt to assess and document the endpoint security characteristics of each part of the taxonomy.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

3. Problem Statement

User Equipment encompasses a very broad set of endpoints. It may be useful to provide a set of categories - or, groups - of endpoints that have similar properties. Endpoints in the same groups may share security characteristics that are particular to that group. The fundamental question is: can a classification of endpoint devices be created that allows for grouping of endpoints that have similar security characteristics? And: is such a grouping - along with operational experience on the Internet - useful in guiding future security protocol design?

If it is possible to answer each of those questions in the affirmative, then operational experience and research can be done on the security characteristics of each category and influence the development of protocols that have the greatest impact for those type of devices.

4. Simplified Endpoint Schematic

A simplified representation of an endpoint is possible by making the following generalization:

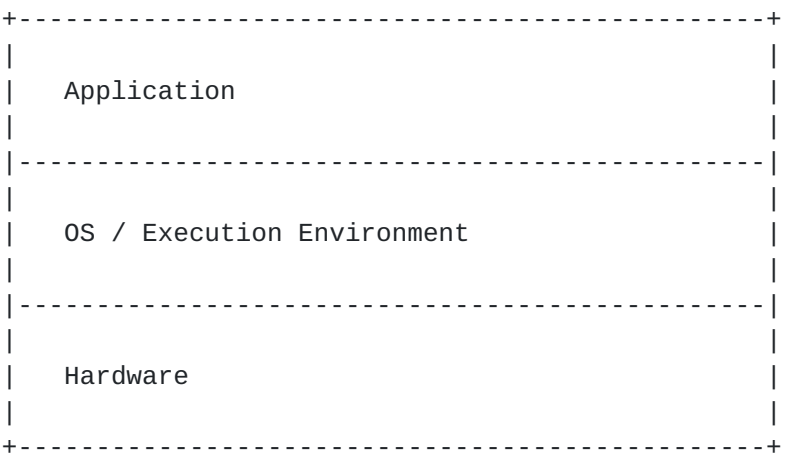

Figure 1 Endpoint Generalization

This simplification means that there are many combinations of hardware, operating systems, execution environments and applications. It also means that any of these three layers can be an endpoint for the purposes of a discussion of endpoint security.

It is natural to suggest that we consider endpoints including those which have a variety of power, computational, storage and network capacities. It is possible that grouping devices with similar

characteristics will help in identifying categories of devices that share similar endpoint security characteristics.

5. Taxonomy and Hierarchy

One suggestion for the taxonomy for endpoints is to consider a hierarchy of endpoints that collects similar endpoint types in large categories and then distinguishes between them in "sub-groups" or lower levels of the taxonomy.

These groupings may provide a way to categorize threats and mitigations to large classes of endpoints on the Internet while providing the ability for differentiation. An example might be a class of endpoints characterized as "constrained devices."

As an example, "constrained devices" might be further subdivided into sub-classes such as sensors, embedded processors, specific (or, special) purpose single-use processors, mesh gateways, and so forth. It can even be imagined that the second level of the hierarchy could be further subdivided by further distinguishing the endpoint types.

The current version of the draft does not take this approach. One of the goals of the endpoint taxonomy is to provide enough differentiation and specificity to ensure that a later operational experience and research can successfully discuss common threats and mitigations for each of the categories in the taxonomy. By providing a ever greater hierarchy of endpoint types, it becomes difficult to scale a future document that discusses threats and mitigations to the highly specific endpoint types.

6. Taxonomy

Others have attempted to provide general-purpose taxonomy and device classification guides. In some settings automated detection and classification of devices provides an essential step in providing appropriate access control and security services.

General-purpose classification systems tend to ossify or become enormously complex. Classification has come from commercial entities, computer science organizations, the academic community and even regional collections of cooperating national governments.

Because of this, we limit the discussion to a taxonomy for endpoints only. We divide endpoints into eight different classes and then attempt to carefully describe the characteristics of devices in each class.

6.1. Traditional and Enterprise Computing Equipment [TECE]

6.1.1. Description

Traditional and Enterprise Computing Equipment is characterized by its extremely high-capacity for transactional volume, storage and shared user population. TECE forms the backbone of high-volume, high-availability transactional computing and is provided in both physical and virtualized forms.

Traditional computing endpoints are shared computing environments characterized by centralized, shared computing. These endpoints are often in large scale data centers. These endpoints are capable of high-availability, substantial requirements for power and environmental control. These endpoints are also characterized by very complex operating systems and user environments.

6.1.2. Endpoint characteristics

- Cost - these endpoints are characterized by extremely high cost.
- Physical size - these are very large endpoints, not suitable or intended for use by an individual.
- Network link characteristics - capable of supporting extremely high bandwidth.
- User interface - very complex and shared among multiple individuals.
- Processing power - extremely high processing capability.
- Physical power - requires substantial provision of electrical power and environmental controls.
- Code complexity - Extremely high support for very complex code including parallelism, multitasking and multithreaded execution.

6.2. Personal Computing Equipment

6.2.1. Description

These are endpoints designed or intended to be used by an individual. They can be delivered as fixed, portable or virtual instantiations of the endpoint. It should be noted that virtual instantiations of endpoints introduce complexities in defining the characteristics of the endpoints. In each case, the device supports a mechanism for

human-interface and has the capability for both local storage and processing. The personal computing equipment class is also characterized by relatively low cost and power requirements.

This class of endpoint is also characterized by the devices supporting multiple purpose use. This class is divided into two sub-classes: fixed and mobile endpoints. The mobile subclass is further divided into four other subclasses: laptops, tablets, intelligent phones, and ultraportable personal computing equipment.

Personal computing endpoints usually have at least one, and often many, network links - often supporting a variety of network connectivity technologies. These endpoints are also characterized by having a human interface - either integral to the computing device itself or supplied externally to the computing device.

6.2.2. Endpoint characteristics

- Cost - these endpoints have a huge range of costs, from extremely inexpensive for simple "personal computer on a board" endpoints to moderately expensive for specially configured laptop and fixed devices.

- Physical size - the physical size of these devices range from handheld to a small cabinet for fixed, desktop units.

- Network link characteristics - personal computing endpoints are often characterized by supporting multiple connectivity technologies.

- User interface - personal computing endpoints are characterized by having user interfaces designed for an individual. The interface varies from simple, text-based interaction to gesture, touch and voice control.

- Processing power - these endpoints are characterized by a significant range of processing power: from single CPU units to endpoints that can support multiple concurrent processes.

- Physical power - personal computing endpoints are characterized by using either traditional mains power or power supplied by a battery.

- Code complexity - personal computing endpoints support complex code and often parallel and multithreaded execution of code.

6.3. Human Interface Devices

6.3.1. Endpoint description

Human interface transactions begin with a task-related goal for a user. This leads to a user behavior (such as pointing, typing or touching) which occurs in the current computing environment. The user's action then should trigger an event in the current computing environment.

Early computer science research breaks the taxonomy for Human Interface Devices into four large categories: input devices, pointing devices, indirect pointing and speech recognition. More recent research adds neural interfaces, VR sensors, and human attribute sensors. In all of these cases, the endpoints have the goal of providing a mechanism for user navigation, interconnection, form filling, menu interaction, data entry or sensing of human input (although not to be confused with the following category in the taxonomy). The result is that this category of the taxonomy has been characterized by extremely limited computing capability in the past. In contemporary networks the human interface devices are far more complex and, as a result, subject to a wider collections of risks as endpoints.

Since human interface devices are often the mechanism that provides control of a computing resource, attacks on those devices are of particular concern. In the past, the idea that there was an external threat to a mouse or a pointing device would be ignored. In contrast, today's voice actuated input devices and VR interfaces are sophisticated enough to represent a real platform for attack.

6.3.2. Endpoint characteristics

- Cost - these endpoints are typically low in cost compared to traditional computing equipment. They are often closer in cost to simple peripheral equipment rather than endpoints that provide general purpose computing platforms.

- Physical size - these devices are meant to provide a human interface and are sized appropriately to that use case. Examples include those devices that are small enough to be handheld or worn.

- Network link characteristics - human interface devices are connected in a variety of ways. Early devices were wired to the device to which they provided connectivity. More recently, these devices have a network connection between them and the connected device. Examples of this connection use Bluetooth or other, very local network connections. These devices may have connections to wider networks to support applications such as augmented reality.

- User interface - generally these devices provide a user interface rather than having a distinct user interface of their own. More complex human interface devices have limited interfaces for settings and control of the device, and its connectivity and function.

- Processing power - these devices are characterized by having limited processing power.

- Physical power - most human interface devices are characterized by having limited power requirements. They are sometimes powered by their connection to the device. In other cases, they are powered by a battery.

- Code complexity - human interface devices tend to have either no or very limited capabilities to execute code. Modern interface devices which support presentation of a virtual physical environment are capable of executing the code needed to provide the interface between the presentation of visual (and other) stimuli while responding to gestures and movement of the person using the device.

6.4. Human Sensor Devices

Description

These are endpoints whose primary purpose is to sense, store, transmit or process information about a human being. These endpoints are characterized as having use cases in health and wellness monitoring, human performance enhancement, personalized medicine and human safety.

The endpoints are characterized as sensor devices with the capacity to sense, store and report on data collected on an individual. The sensor may be multimodal. These endpoints are almost always characterized by have a battery for power and having limited storage, networking and processing capabilities.

6.4.1. Endpoint characteristics

- Cost - Human Sensor Endpoints can range in cost from very low (for instance a heartbeat sensor) to quite expensive (a sensor built into an implanted device).

- Physical size - human sensors are very small and almost always portable.

- Network link characteristics - human sensors usually have a single network like technology available and are capable of very limited bandwidth utilization on that link.

- User interface - human sensors have extremely limited, or no, user interface.

- Processing power - human sensors are characterized by having limited processing power - often incorporating only the ability to collect store and forward sensed information.

- Physical power - human sensors are characterized by being powered by internal batteries

- Code complexity - human sensors are not usually capable of running complex code. Often, the capability of the endpoint is to simply sense, store and forward data without reporting and analysis of that data.

6.5. Non-human Sensor Devices

6.5.1. Endpoint Description

These endpoints are capable of sensing, storage, communication and possibly some computation. They are characterized by having very low bandwidth radios, a battery for power, sensor technology and a small processor. Unlike in Section 5.4, these devices are not intended to sense human-related information.

Compared with Human Sensors, non-human sensors often have a variety of communications technologies available - for instance, self-organizing into mesh networks.

6.5.2. Endpoint characteristics

- Cost - Non-human Sensor Endpoints can range in cost from very low (for instance, a simple temperature sensor) to quite expensive (a sensor built into an implanted device.

- Physical size - Non-human sensors are often small and almost always portable.

- Network link characteristics - Non-human sensors usually have a single network like technology available but the topology of those network links can be highly varied. Quite often these devices are capable of very limited bandwidth utilization on the link to which they are attached.

- User interface - non-human sensors have extremely limited, or no, user interface.

- Processing power - non-human sensors are characterized by having limited processing power - often incorporating only the ability to collect store and forward sensed information. Some non-human sensors have the capability to process stored data, but usually this is limited.

- Physical power - non-human sensors often require very limited amounts of power - very often provided by a battery.

- Code complexity - non-human sensors are not usually capable of running complex code. Often, the capability of the endpoint is to simply sense, store and forward data without reporting and analysis of that data.

6.6. Peripheral Computing Equipment and Embedded Endpoints

6.6.1. Endpoint Description

These are endpoints that are "embedded" in devices that may have a different primary function. An example is a network endpoint in a printer that supports remote access, configuration and printing. Another example is an endpoint in an appliance that has a different primary function (for instance, a refrigerator).

In either case, the endpoint is characterized as being added to another system, machine or peripheral.

These devices are characterized as being specialized for their particular use case and function. Their specific characteristics often depend upon the system, device or peripheral in which they are being hosted. As an example, the embedded endpoint gets its physical power and networking capabilities from the device in which it is connected.

6.6.2. Endpoint characteristics

- Cost - almost never available as a standalone device - instead, always embedded into the peripheral or system which is hosting it.
- Physical size - almost always very small - to be embedded into some other system or device.
- Network link characteristics - dependent on network services available from the host device and not always IP-based.
- User interface - almost always provided by the "hosting" device. Many embedded endpoints share a user interface with the configuration and control tool for the underlying device.
- Processing power - usually limited and constrained by the use case. Some embedded endpoints provide remote access to the underlying resources provided by the processor.
- Physical power - generally supplied by the "host" system or device.
- Code complexity - limited and almost always constrained by use case.

6.7. Application Layer Endpoints

6.7.1. Description

A significant trend in the contemporary public Internet is to have applications act as completely independent agents - a situation where the application itself provides the necessary infrastructure (for instance, domain name resolution) to provide services. An example would be a web browser that independently resolved domain names and established secure communication channels independently.

The traffic between the application and the servers it uses might not be available for analysis by security software. As a result, application-based endpoints would have the characteristic of having to provide security services (for instance, traffic security or malware detection) for itself.

This type of endpoint also has the characteristic of potentially having adverse impacts on other applications running on the same platform. For example, if several applications are provisioning their own infrastructure services, then those services are being duplicated on that platform. For security related infrastructure there would be

no common, platform-wide approach to securing the applications or the traffic generated between the application and external servers.

6.7.2. Endpoint Characteristics

- Cost - applications vary widely in cost and some are free.
- Physical size - based on code, application endpoints do not have physical characteristics (e.g. size, power requirements, etc.).
- Network link characteristics - applications often use network facilities provided by lower layers of the stack. In particular, many application endpoints use the network services provided by the underlying operating system that acts as the host for the application. An emerging trend in both wired and wireless networks is for the application to interface with the network link to control or provide some of the network link services for itself. An example of this would be an application that does DNS resolution services for itself rather then depending on the underlying operating system to provide that service.
- User interface - the application usually provides its own user interface which can be minimal (for instance, command line driven) or complex (windows or VR driven).
- Processing power - always dependent on the device on which the application is hosted.
- Physical power - based on code, application endpoints do not have physical requirements (e.g. power)
- Code complexity - highly variable. Applications can be very simple or highly complex depending on the application's requirements.

6.8. Edge Network and Acquisition Endpoints

6.8.1. Description

The emergence of intelligent devices and things has led to new network designs where data is aggregated at points topologically close to where the data is gathered. The gathered data can then have the option to flow to nearby gateways, or a Wi-Fi/W-LAN (SD-WAN) router/equipment, or the telco tower/rooftop towers. These often perform an acquisition function that includes both aggregation and data condensation.

They usually have some level of processing capability. The main task for these devices is to collect the data from various other endpoints and send the processed data upstream. In doing so, they often perform some low-level data processing, such as data filtering (which determines what data is sent/blocked) and data analytics.

The acquisition systems are often architected to talk to distributed data centers and end devices; for instance, on a factory shop floor, a CDN's edge PoP (Point of Presence), an edge colocation local, or a metro regional datacenter for a Telco or IT Service Provider.

In all cases, these edge computing devices represent a newer class of endpoints. These are endpoints that are not at the extreme edge of the network, but provide services to the devices at those edges (especially for those devices in the class discussed in section 6.4 and 6.5 above).

The threats and mitigations for this class of device is expected to be significantly different from those in sections 6.4 and 6.5.

6.8.2. Endpoint characteristics

- Cost - highly variable. Edge network devices in 5G networks can be very expensive. Aggregation nodes in sensor networks can be very inexpensive.

- Physical size - highly variable. Edge network devices in 5G networks can be larger than personal computing equipment. Aggregation nodes in sensor networks can be as small as a circuit board, battery and radio.

- Network link characteristics - by their nature, these devices have at least a pair of network links. One of these links faces toward the network where the data is being aggregated. The other faces toward the network where the data is being processed, analyzed or reported upon.

- User interface - these devices usually have a limited user interface, characterized by the need to configure the device, provide security and allow for management of the network links.

- Processing power - usually these devices have limited processing power: their emphasis is on aggregation and management of data flows between networks.

- Physical power - highly variable. Edge network devices in 5G networks can require significant sources of secure and consistent power. Aggregation nodes in sensor networks can often be supported by a small battery.

- Code complexity - usually these devices have limited ability to load and execute code. Since their emphasis is on aggregation and management of data flows between networks, these devices usually have minimal ability to run general purpose code.

7. Security Considerations

This draft is non-normative and simply attempts to provide a taxonomy for endpoints. The goal of the taxonomy is to document that there are classes of endpoints that have different characteristics. Those classes may have completely different threat landscapes and the endpoints may have completely different security capabilities.

This document is intended to support further work in that combines operational security experience with guidance for security protocol design.

8. IANA Considerations

This document has no requirements or actions for IANA.

9. References

9.1. Normative References

[1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

9.2. Informative References

[I-D:draft-taddei-smart-cless-introduction] Taddei, A., Wueest, C., Roundy, K., Lazanski, D., "Capabilities and Limitations of an Endpoint-only Security Solution," https://tools.ietf.org/html/draft-taddei-smart-cless-introduction-01, March 2020.

10. Acknowledgments

This document was prepared using 2-Word-v2.0.template.dot.

Appendix A. Document History

-00

Initial Internet Draft

Authors' Addresses

Mark McFadden
Internet policy advisors ltd
Chepstow Wales UK

Email: mark@internetpolicyadvisors.com