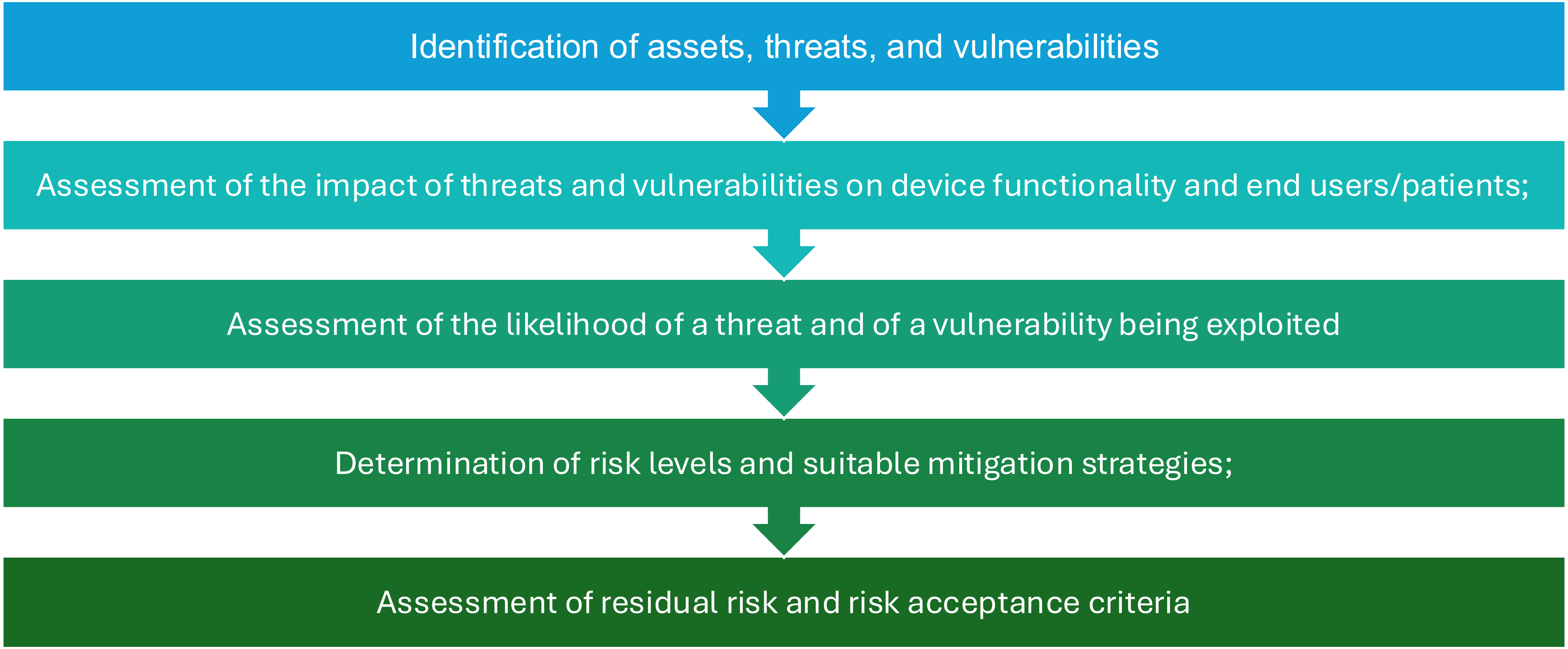
DATA PROTECTION AND CYBERSECURITY IN eHEALTH SYSTEMS 3

1924 2024

UNIVERSITÀ DEGLI STUDI DI TRIESTE

Cybersecurity risk assesment



Identification of assets, threats, and vulnerabilities

Assessment of the impact of threats and vulnerabilities on device functionality and end users/patients;

Assessment of the likelihood of a threat and of a vulnerability being exploited

Determination of risk levels and suitable mitigation strategies;

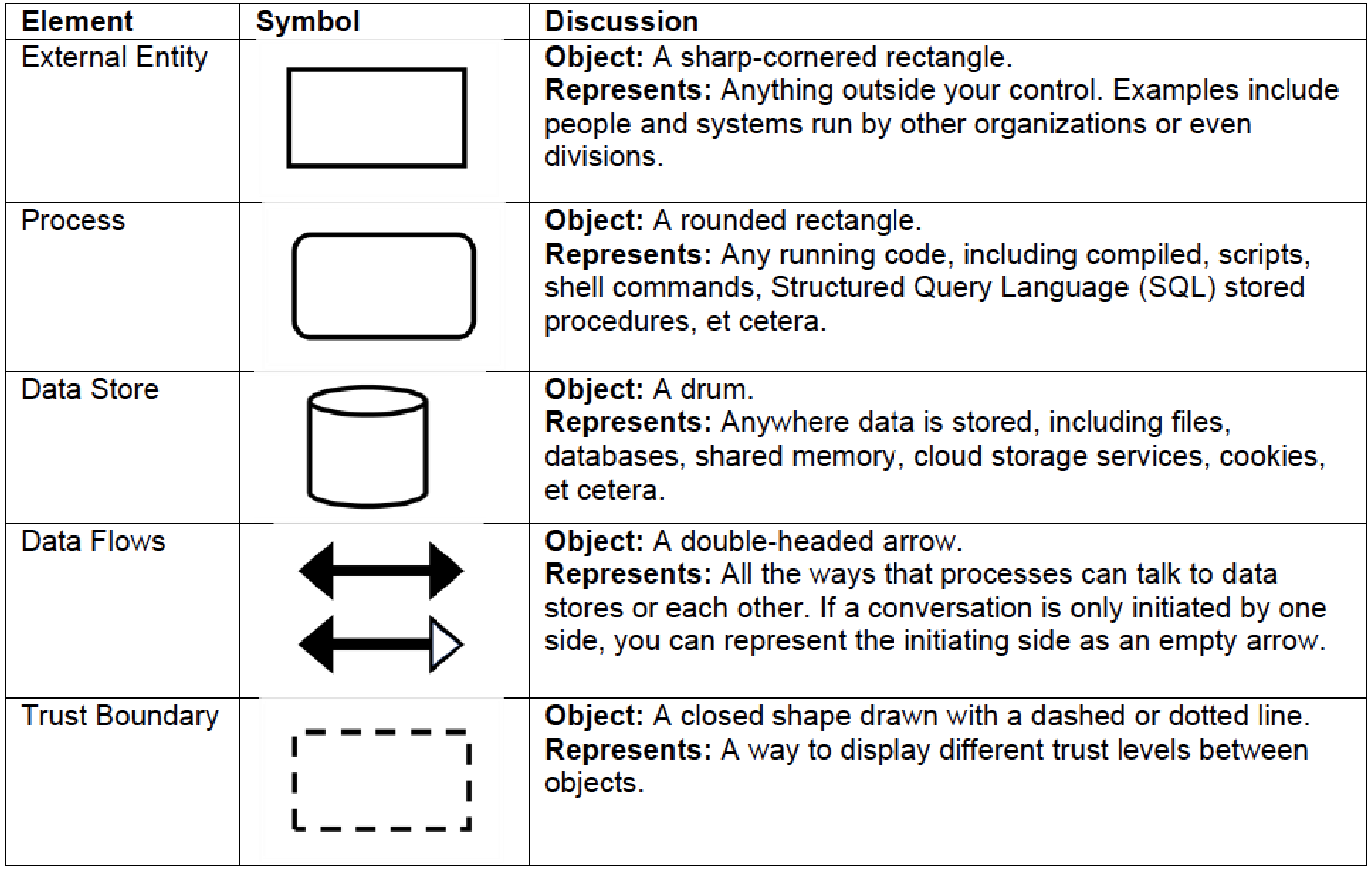
Assessment of residual risk and risk acceptance criteria

Identification of threats step 1. System Modeling

- Formal process and system modeling helps identifying threats and vulnerabilities
- Multiple techniques:
  - UML
  - Data flow diagrams (DFDs)

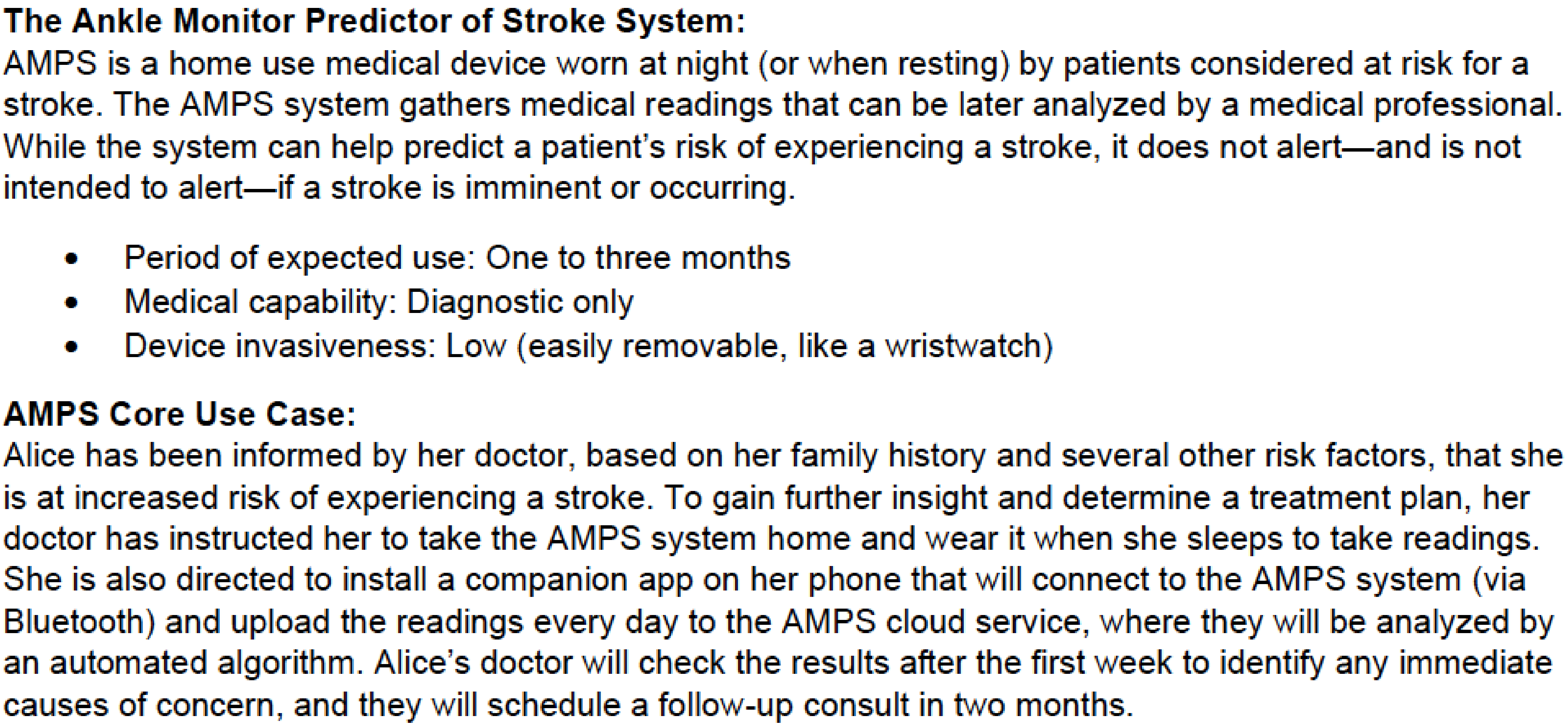
Data flow diagram

DFDs are a way to represent the entities involved with the functioning of the medical device, how those entities are related, and the assumed trust boundaries between them.



Element	Symbol	Discussion
External Entity	▭	**Object:** A sharp-cornered rectangle. **Represents:** Anything outside your control. Examples include people and systems run by other organizations or even divisions.
Process	▢	**Object:** A rounded rectangle. **Represents:** Any running code, including compiled, scripts, shell commands, Structured Query Language (SQL) stored procedures, et cetera.
Data Store	⛁	**Object:** A drum. **Represents:** Anywhere data is stored, including files, databases, shared memory, cloud storage services, cookies, et cetera.
Data Flows	⟷ ⟷	**Object:** A double-headed arrow. **Represents:** All the ways that processes can talk to data stores or each other. If a conversation is only initiated by one side, you can represent the initiating side as an empty arrow.
Trust Boundary	⬚	**Object:** A closed shape drawn with a dashed or dotted line. **Represents:** A way to display different trust levels between objects.

Example: definition of the system and use cases

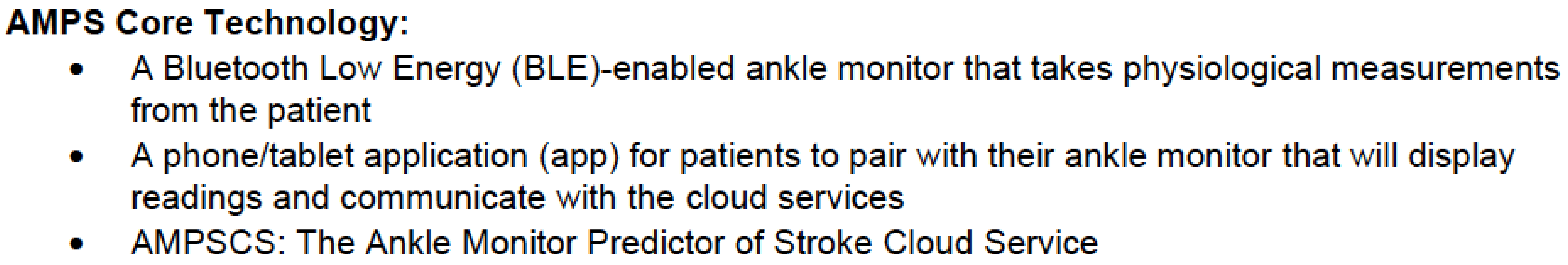


The Ankle Monitor Predictor of Stroke System:
AMPS is a home use medical device worn at night (or when resting) by patients considered at risk for a stroke. The AMPS system gathers medical readings that can be later analyzed by a medical professional. While the system can help predict a patient's risk of experiencing a stroke, it does not alert—and is not intended to alert—if a stroke is imminent or occurring.

- Period of expected use: One to three months
- Medical capability: Diagnostic only
- Device invasiveness: Low (easily removable, like a wristwatch)

AMPS Core Use Case:
Alice has been informed by her doctor, based on her family history and several other risk factors, that she is at increased risk of experiencing a stroke. To gain further insight and determine a treatment plan, her doctor has instructed her to take the AMPS system home and wear it when she sleeps to take readings. She is also directed to install a companion app on her phone that will connect to the AMPS system (via Bluetooth) and upload the readings every day to the AMPS cloud service, where they will be analyzed by an automated algorithm. Alice's doctor will check the results after the first week to identify any immediate causes of concern, and they will schedule a follow-up consult in two months.

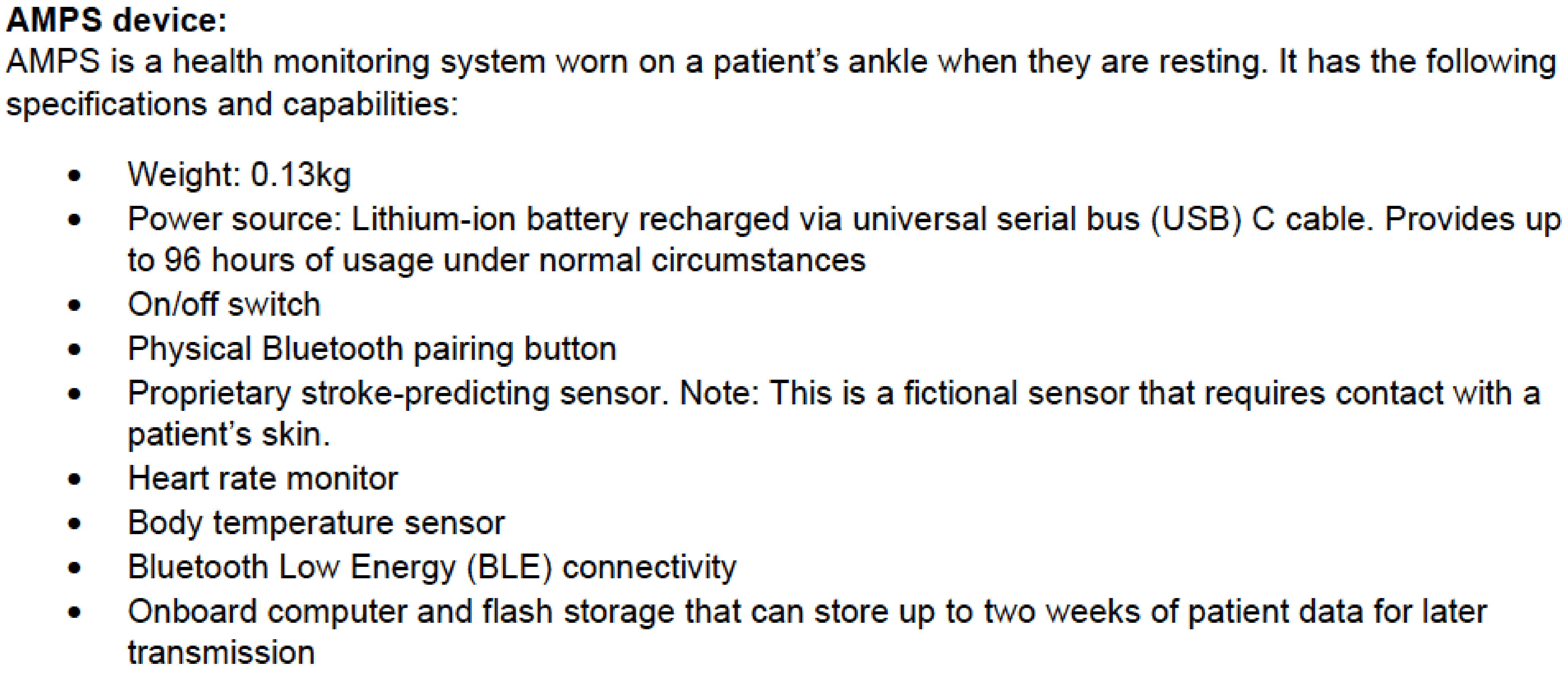
Example: core technology



AMPS Core Technology:

- A Bluetooth Low Energy (BLE)-enabled ankle monitor that takes physiological measurements from the patient
- A phone/tablet application (app) for patients to pair with their ankle monitor that will display readings and communicate with the cloud services
- AMPSCS: The Ankle Monitor Predictor of Stroke Cloud Service

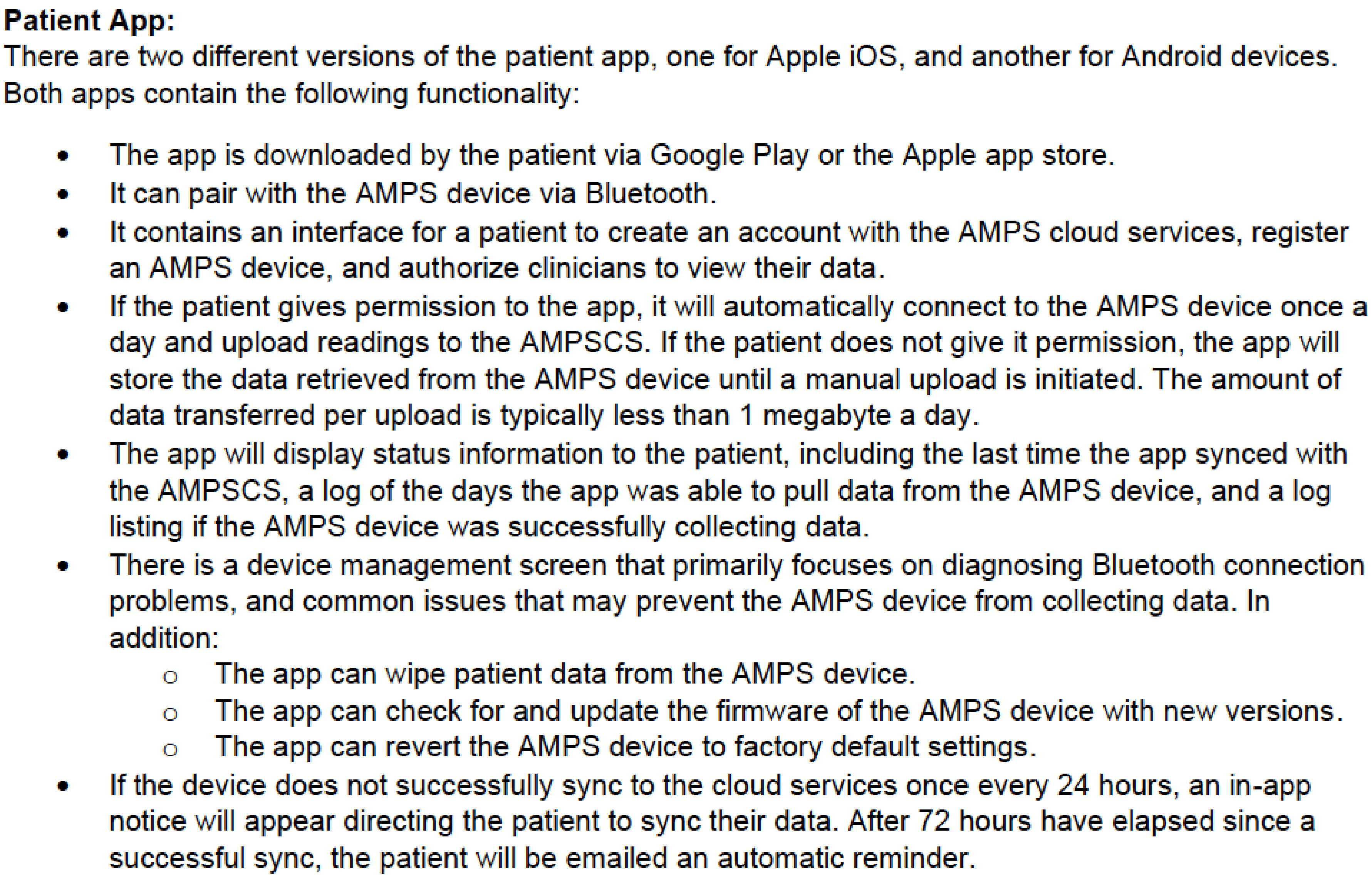
Example: AMPS device



AMPS device:
AMPS is a health monitoring system worn on a patient’s ankle when they are resting. It has the following specifications and capabilities:

- Weight: 0.13kg
- Power source: Lithium-ion battery recharged via universal serial bus (USB) C cable. Provides up to 96 hours of usage under normal circumstances
- On/off switch
- Physical Bluetooth pairing button
- Proprietary stroke-predicting sensor. Note: This is a fictional sensor that requires contact with a patient’s skin.
- Heart rate monitor
- Body temperature sensor
- Bluetooth Low Energy (BLE) connectivity
- Onboard computer and flash storage that can store up to two weeks of patient data for later transmission

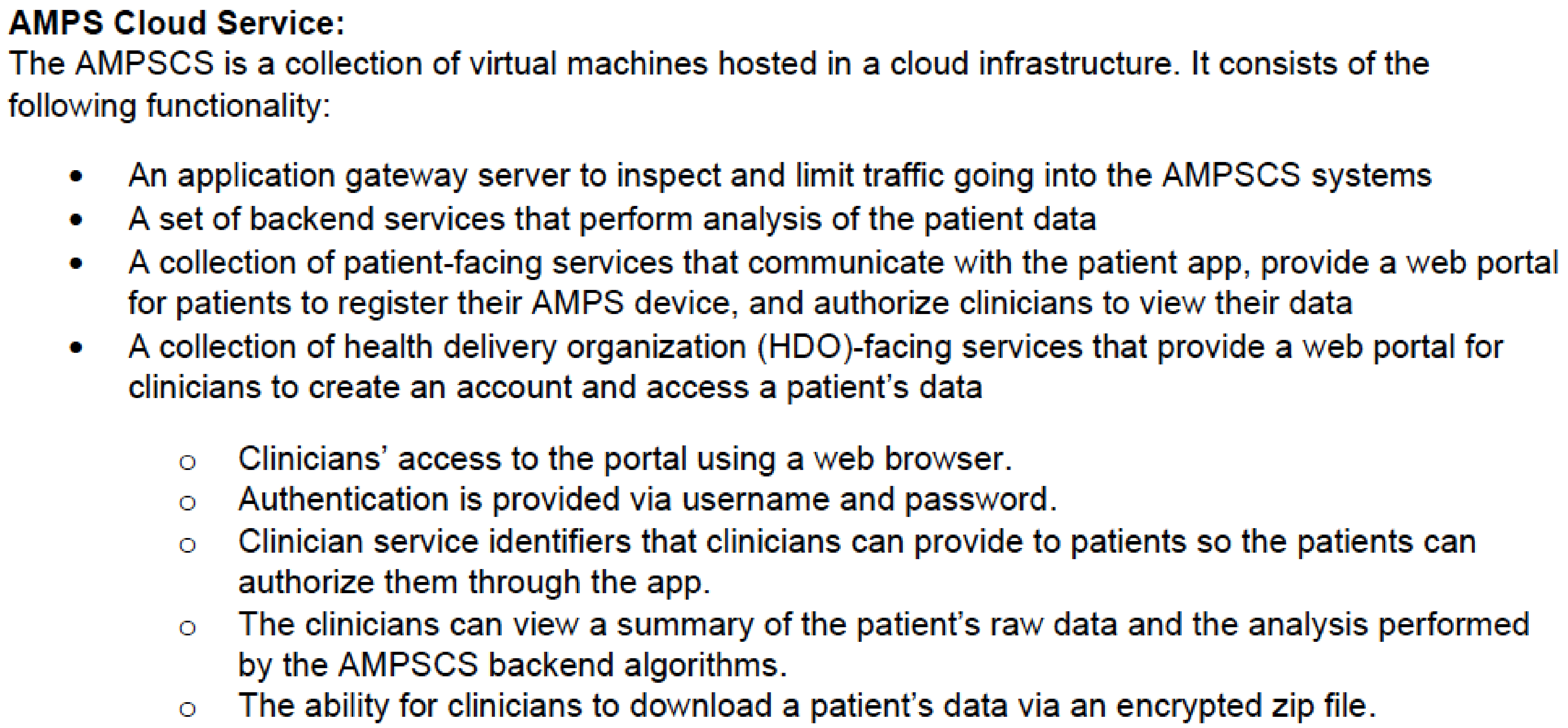
Example: patient app



Patient App:
There are two different versions of the patient app, one for Apple iOS, and another for Android devices. Both apps contain the following functionality:

- The app is downloaded by the patient via Google Play or the Apple app store.
- It can pair with the AMPS device via Bluetooth.
- It contains an interface for a patient to create an account with the AMPS cloud services, register an AMPS device, and authorize clinicians to view their data.
- If the patient gives permission to the app, it will automatically connect to the AMPS device once a day and upload readings to the AMPSCS. If the patient does not give it permission, the app will store the data retrieved from the AMPS device until a manual upload is initiated. The amount of data transferred per upload is typically less than 1 megabyte a day.
- The app will display status information to the patient, including the last time the app synced with the AMPSCS, a log of the days the app was able to pull data from the AMPS device, and a log listing if the AMPS device was successfully collecting data.
- There is a device management screen that primarily focuses on diagnosing Bluetooth connection problems, and common issues that may prevent the AMPS device from collecting data. In addition:
  - The app can wipe patient data from the AMPS device.
  - The app can check for and update the firmware of the AMPS device with new versions.
  - The app can revert the AMPS device to factory default settings.
- If the device does not successfully sync to the cloud services once every 24 hours, an in-app notice will appear directing the patient to sync their data. After 72 hours have elapsed since a successful sync, the patient will be emailed an automatic reminder.

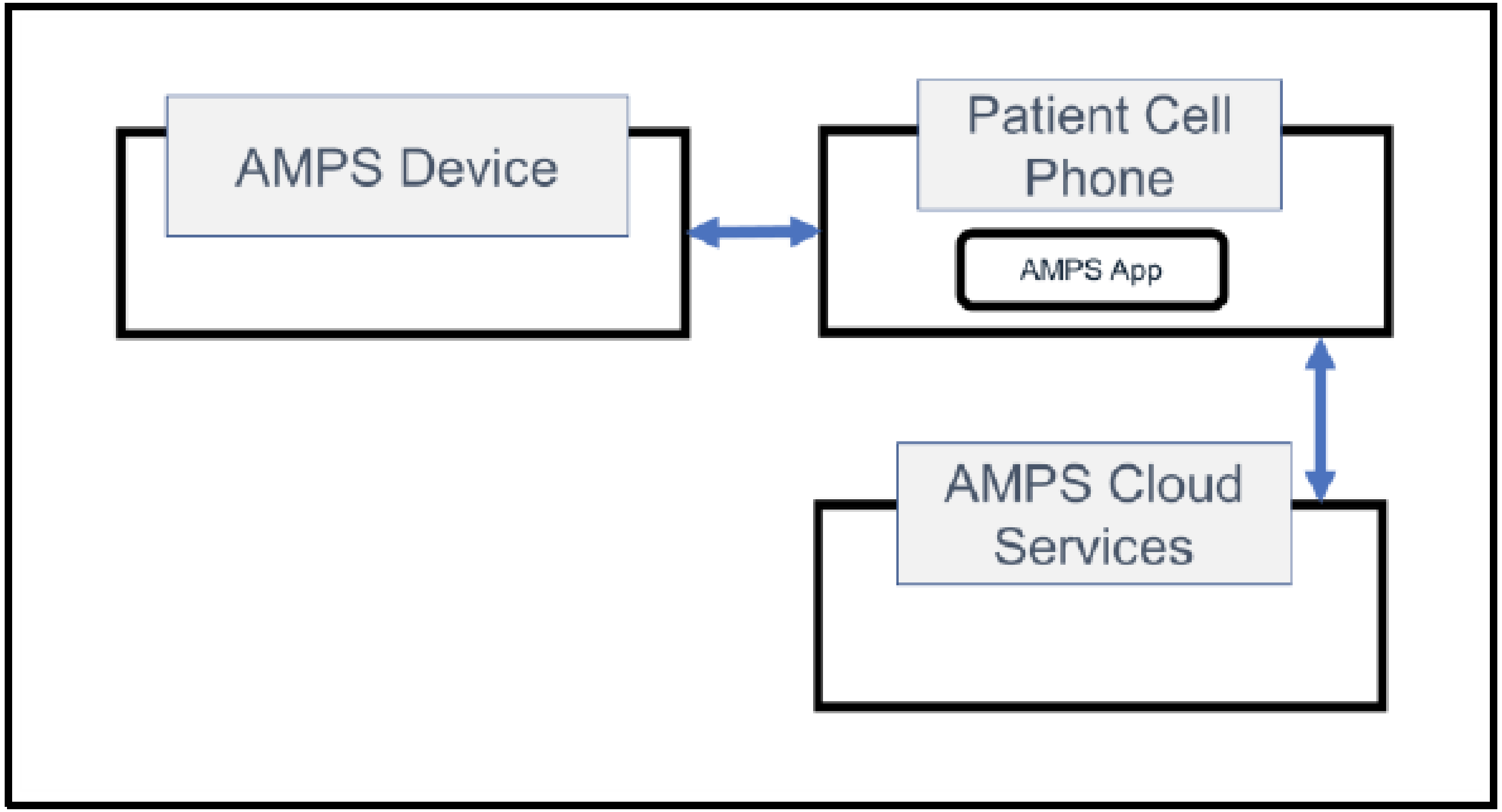
Example: cloud service



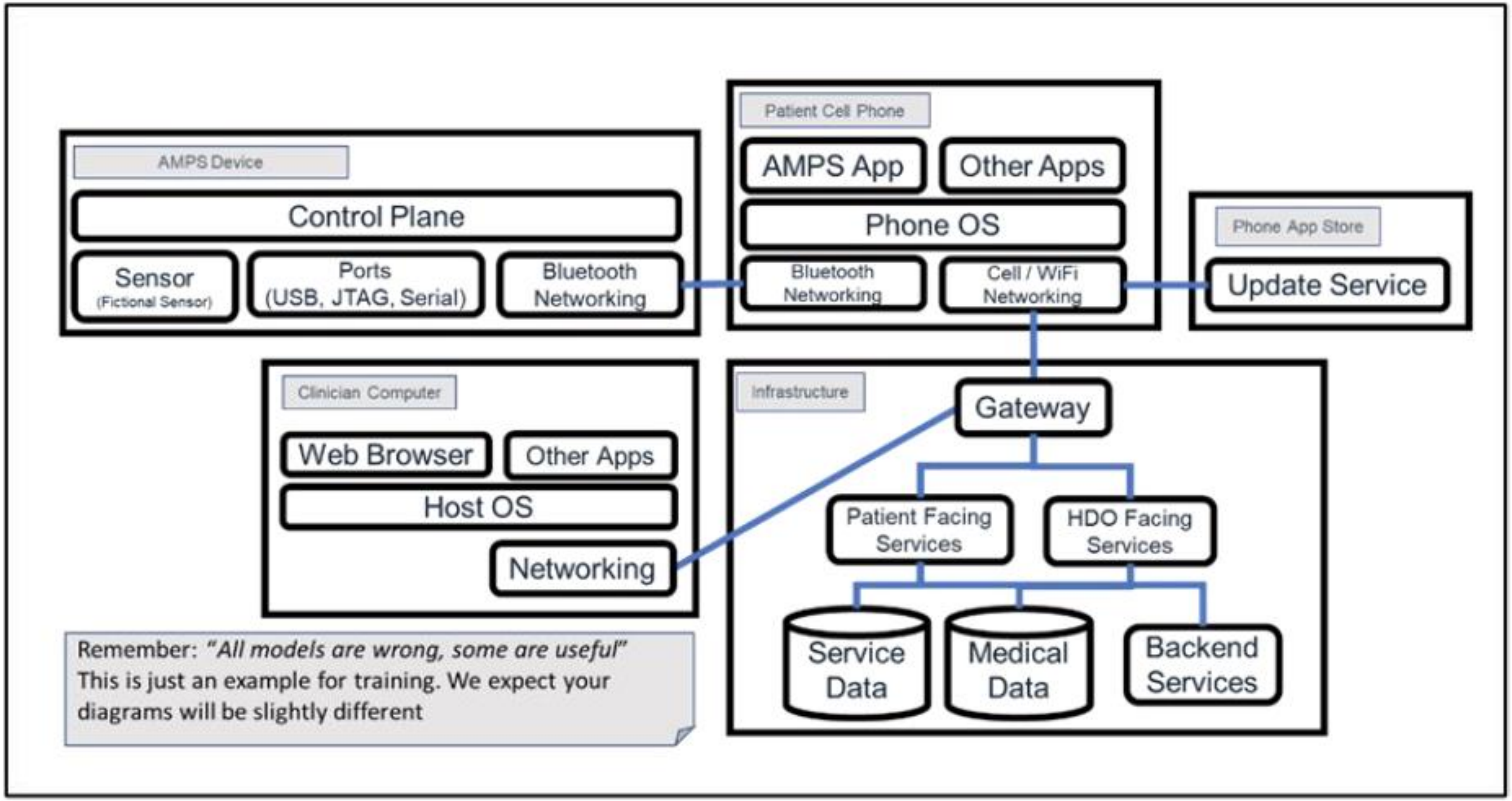
AMPS Cloud Service:
The AMPSCS is a collection of virtual machines hosted in a cloud infrastructure. It consists of the following functionality:

- An application gateway server to inspect and limit traffic going into the AMPSCS systems
- A set of backend services that perform analysis of the patient data
- A collection of patient-facing services that communicate with the patient app, provide a web portal for patients to register their AMPS device, and authorize clinicians to view their data
- A collection of health delivery organization (HDO)-facing services that provide a web portal for clinicians to create an account and access a patient's data
    - Clinicians' access to the portal using a web browser.
    - Authentication is provided via username and password.
    - Clinician service identifiers that clinicians can provide to patients so the patients can authorize them through the app.
    - The clinicians can view a summary of the patient's raw data and the analysis performed by the AMPSCS backend algorithms.
    - The ability for clinicians to download a patient's data via an encrypted zip file.

Example: high-level diagram



Example: detailed DFD

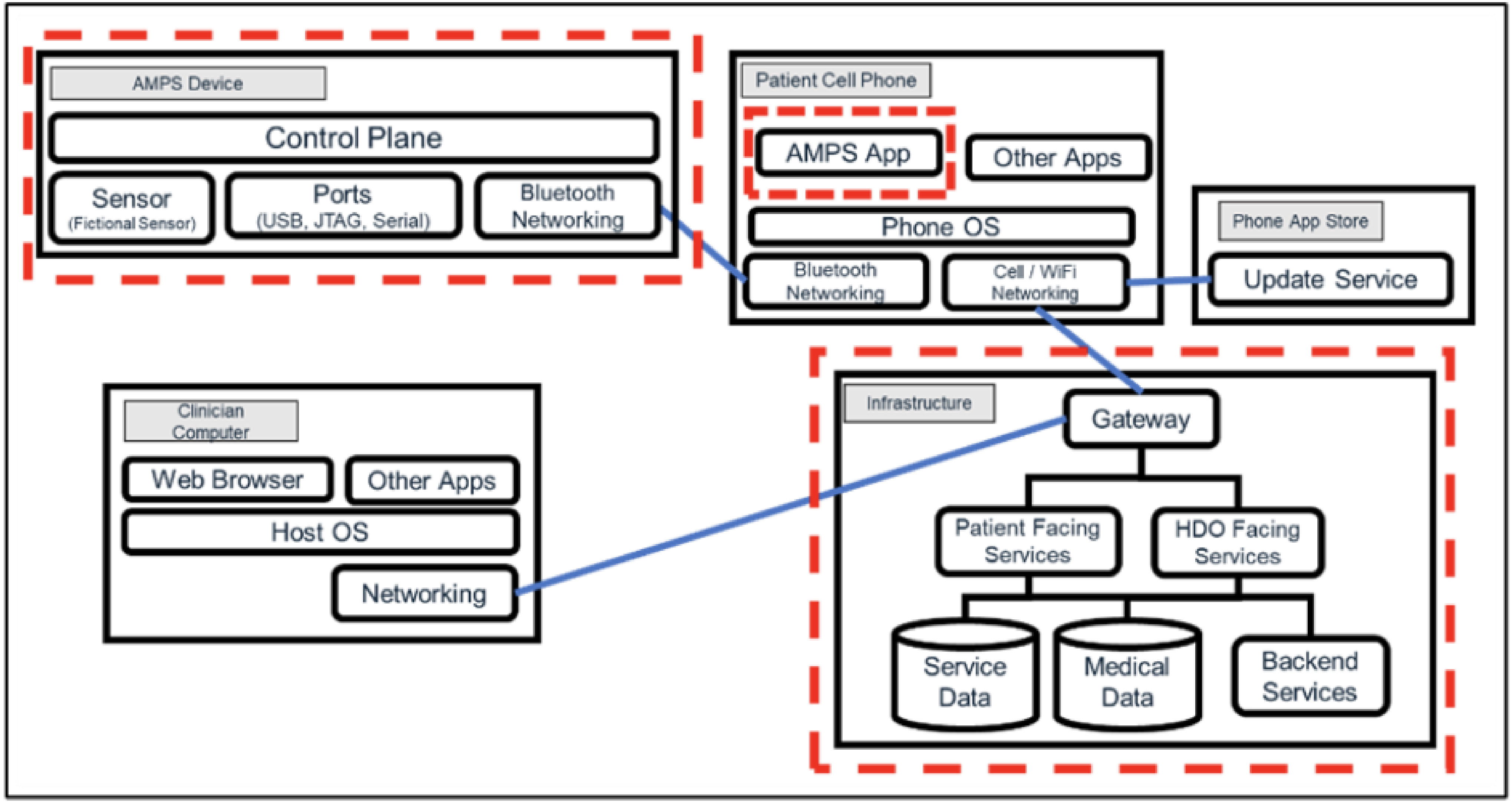


Remember: *"All models are wrong, some are useful"*
This is just an example for training. We expect your diagrams will be slightly different

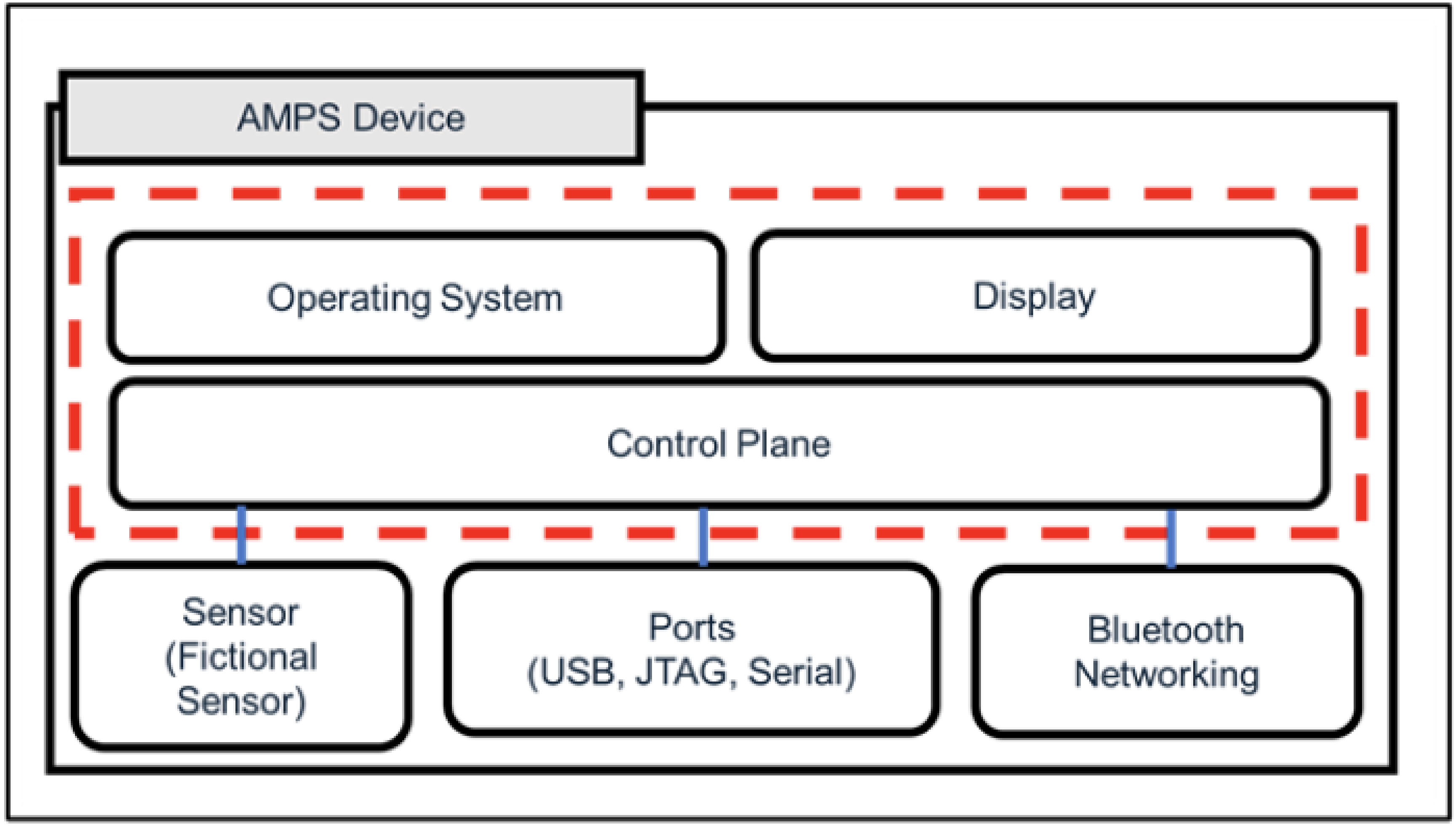
Trust boundaries

- Trust boundaries do not physically reside in a given organization’s system, but instead represent ideas and assumptions being made by the threat modeling team about how different entities interact.
- Trust boundaries help in later stages of the threat modeling process by identifying areas that require enhanced investigation.
- Trust boundaries help capture the thought process of the threat modeling team and can be used to help convey that information to external reviewers.

Example: Trust boundaries



Example: Trust boundaries within a device

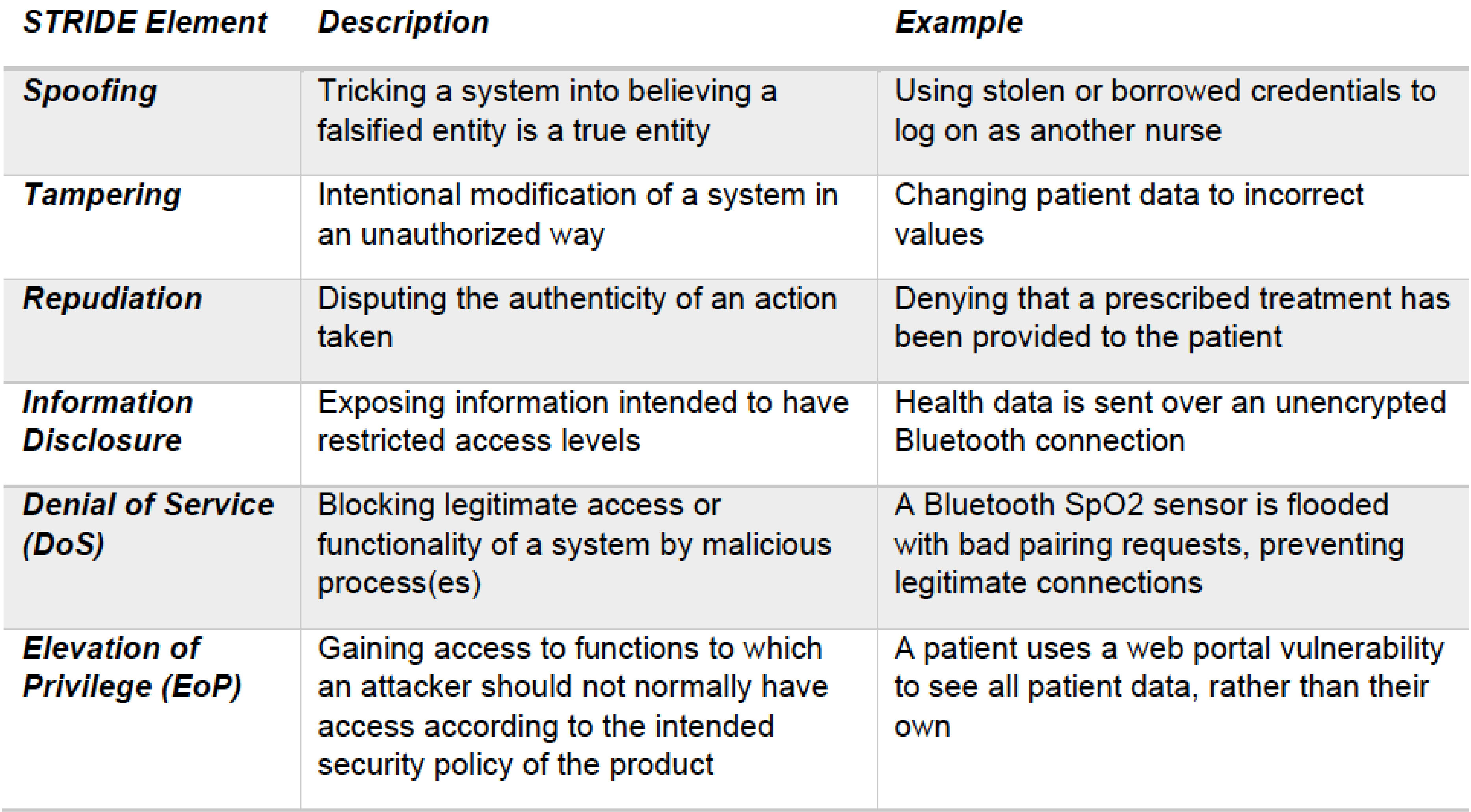


Identification of threats step 2. threat identification

- There are several techniques:
  - STRIDE (Spoofing, Tampering, Repudiation, Information disclosure, Denial of service, Elevation of privilege)
  - Attack trees
  - Kill Chains and Cyber Attack Lifecycles
  - ATT&CK Framework

STRIDE

STRIDE is a mnemonic that articulates six types of potential threats against a system.



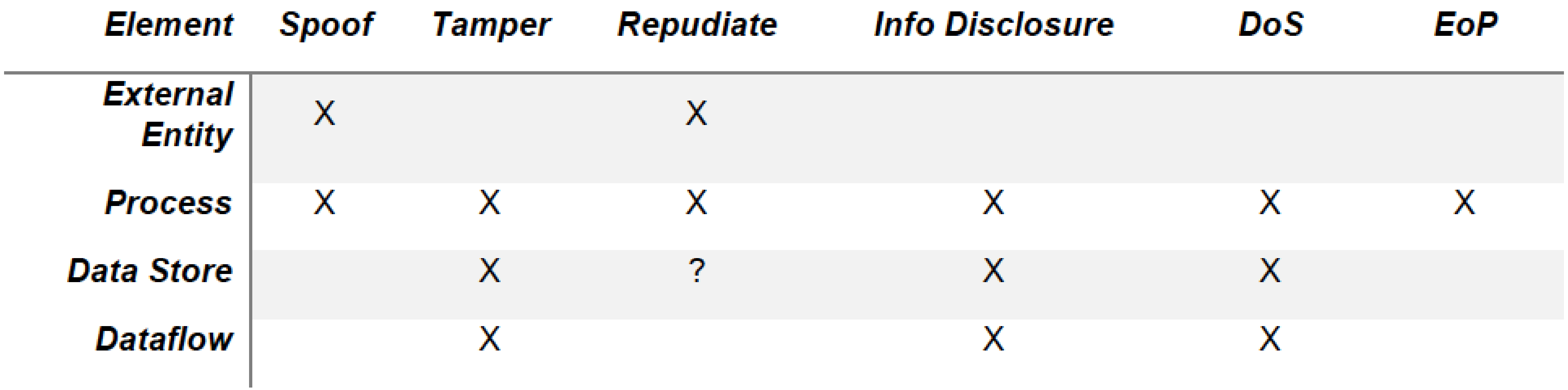
STRIDE Element	***Description***	***Example***
Spoofing	Tricking a system into believing a falsified entity is a true entity	Using stolen or borrowed credentials to log on as another nurse
Tampering	Intentional modification of a system in an unauthorized way	Changing patient data to incorrect values
Repudiation	Disputing the authenticity of an action taken	Denying that a prescribed treatment has been provided to the patient
Information Disclosure	Exposing information intended to have restricted access levels	Health data is sent over an unencrypted Bluetooth connection
Denial of Service (DoS)	Blocking legitimate access or functionality of a system by malicious process(es)	A Bluetooth SpO2 sensor is flooded with bad pairing requests, preventing legitimate connections
Elevation of Privilege (EoP)	Gaining access to functions to which an attacker should not normally have access according to the intended security policy of the product	A patient uses a web portal vulnerability to see all patient data, rather than their own

STRIDE per ELEMENT

STRIDE can be applied to the DFD elements or dataflow (“STRIDE per Element” approach).

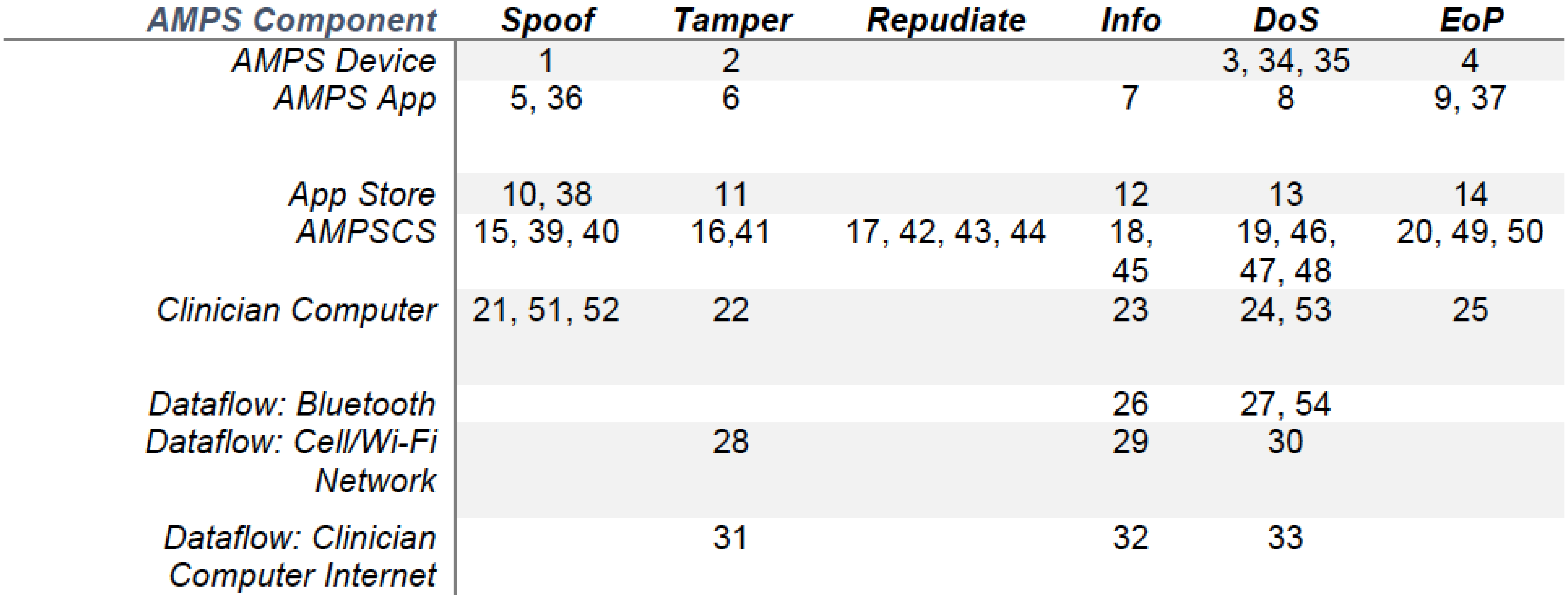
This method is developed by analyzing which STRIDE threats tend to appear for individual DFD element types.

This approach creates a mapping where for a particular DFD element, there will be a list of STRIDE threats commonly associated with it.



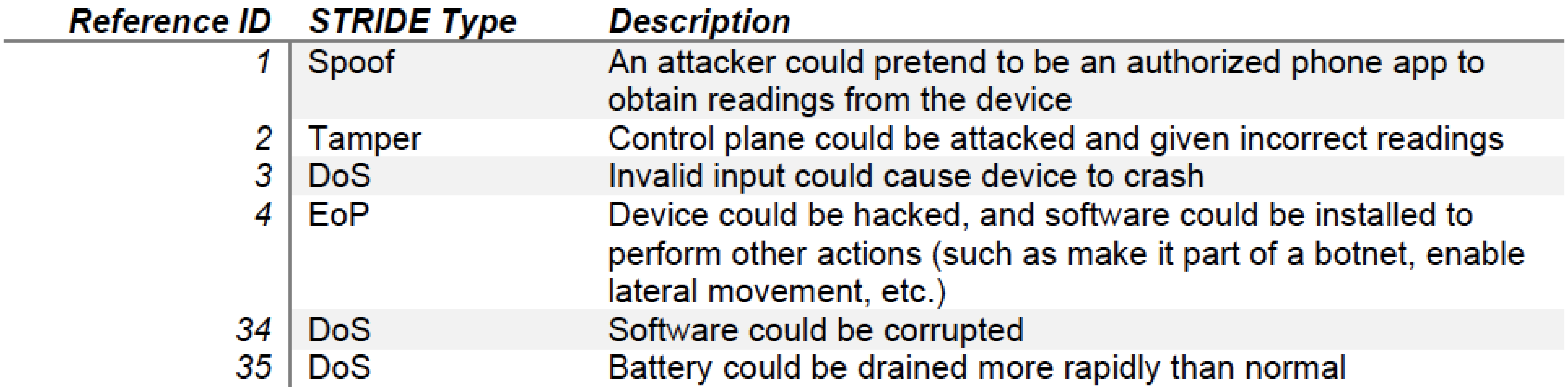
Element	Spoof	Tamper	Repudiate	Info Disclosure	DoS	EoP
External Entity	X		X			
Process	X	X	X	X	X	X
Data Store		X	?	X	X	
Dataflow		X		X	X	

STRIDE per application



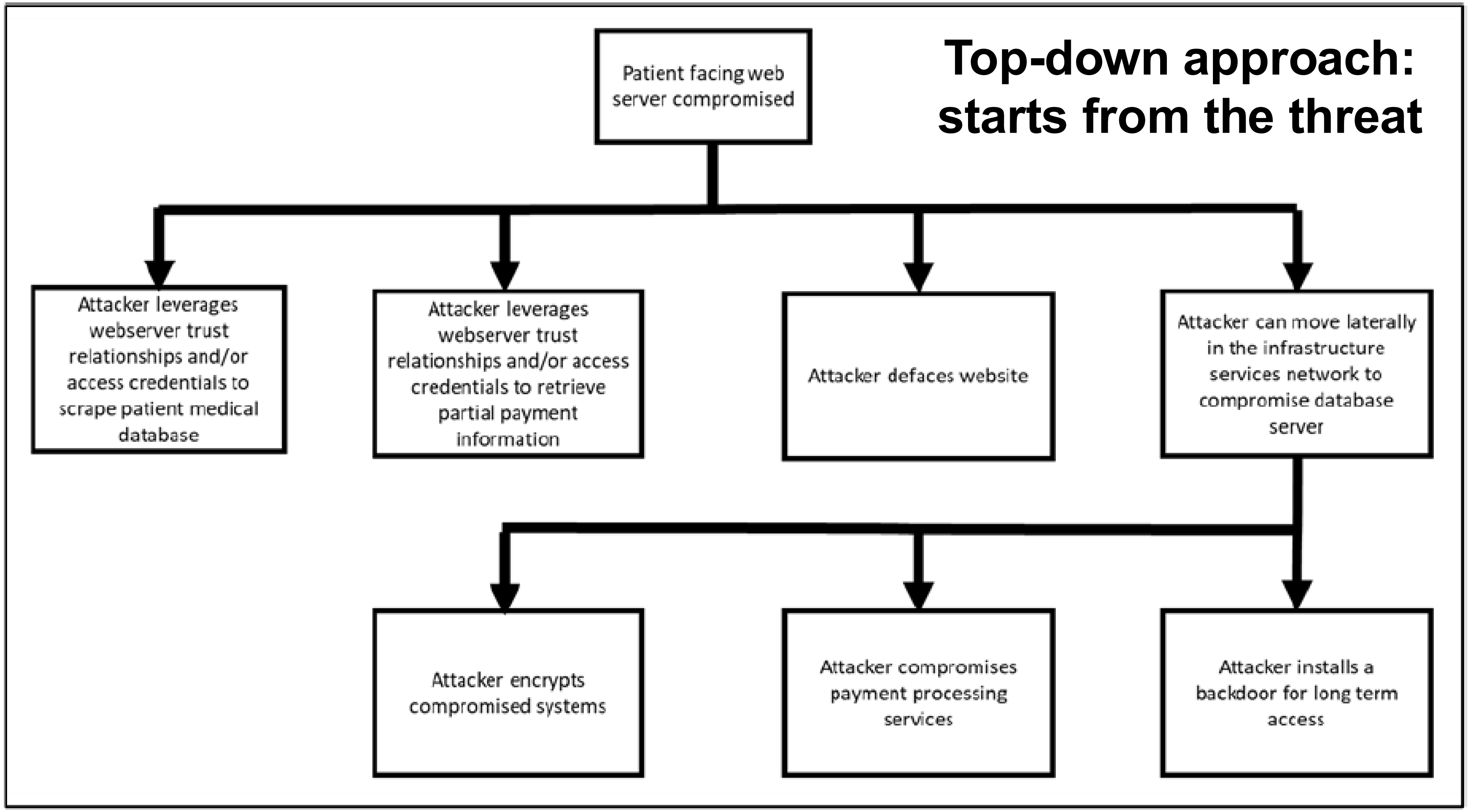
AMPS Component	*Spoof*	*Tamper*	*Repudiate*	*Info*	*DoS*	*EoP*
AMPS Device	1	2			3, 34, 35	4
AMPS App	5, 36	6		7	8	9, 37
App Store	10, 38	11		12	13	14
AMPSCS	15, 39, 40	16,41	17, 42, 43, 44	18, 45	19, 46, 47, 48	20, 49, 50
Clinician Computer	21, 51, 52	22		23	24, 53	25
Dataflow: Bluetooth				26	27, 54	
Dataflow: Cell/Wi-Fi Network		28		29	30	
Dataflow: Clinician Computer Internet		31		32	33	

Example: STRIDE per application



Reference ID	*STRIDE Type*	*Description*
1	Spoof	An attacker could pretend to be an authorized phone app to obtain readings from the device
2	Tamper	Control plane could be attacked and given incorrect readings
3	DoS	Invalid input could cause device to crash
4	EoP	Device could be hacked, and software could be installed to perform other actions (such as make it part of a botnet, enable lateral movement, etc.)
34	DoS	Software could be corrupted
35	DoS	Battery could be drained more rapidly than normal

Attack trees



Top-down approach: starts from the threat

Patient facing web server compromised

Attacker leverages webserver trust relationships and/or access credentials to scrape patient medical database

Attacker leverages webserver trust relationships and/or access credentials to retrieve partial payment information

Attacker defaces website

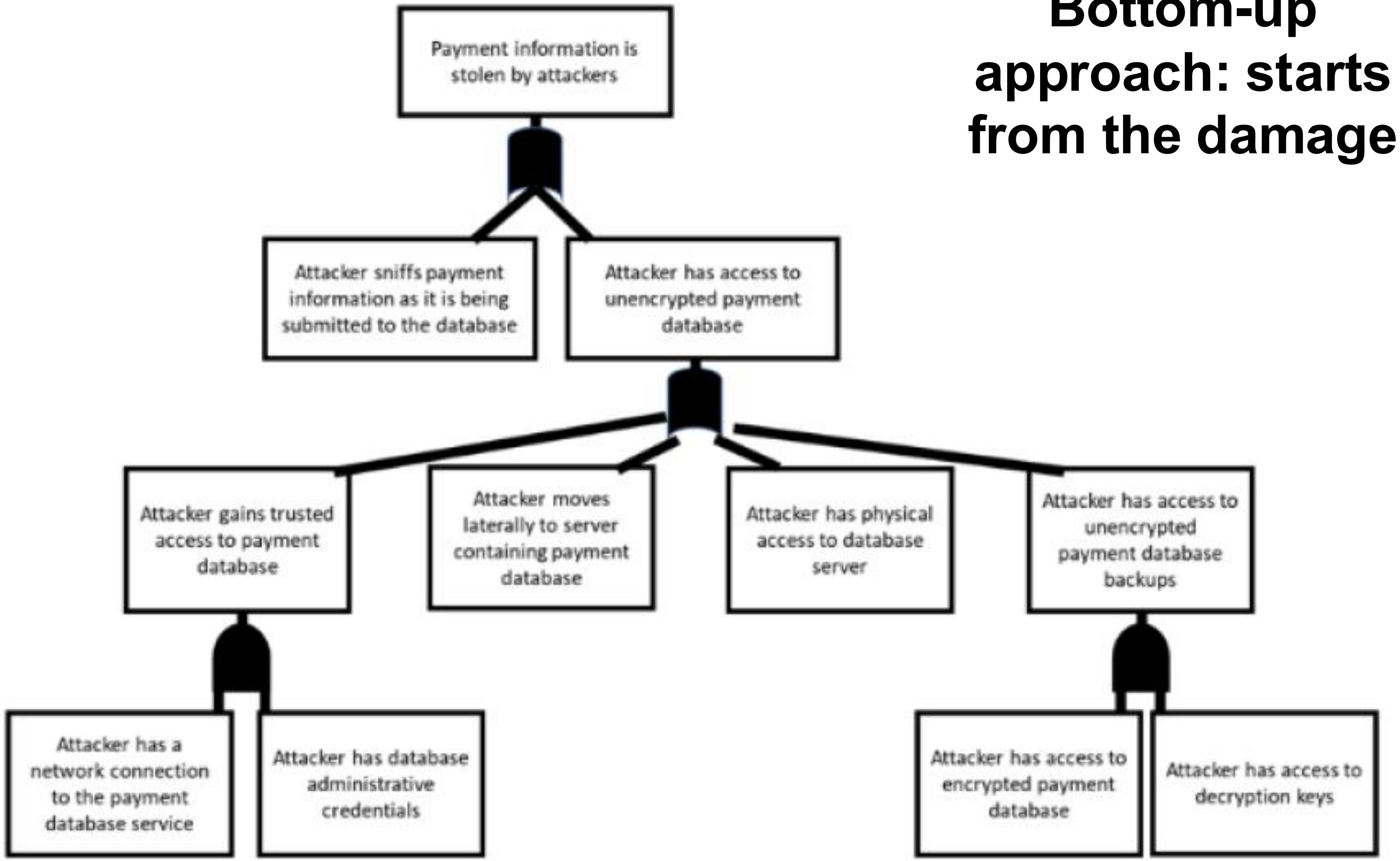
Attacker can move laterally in the infrastructure services network to compromise database server

Attacker encrypts compromised systems

Attacker compromises payment processing services

Attacker installs a backdoor for long term access

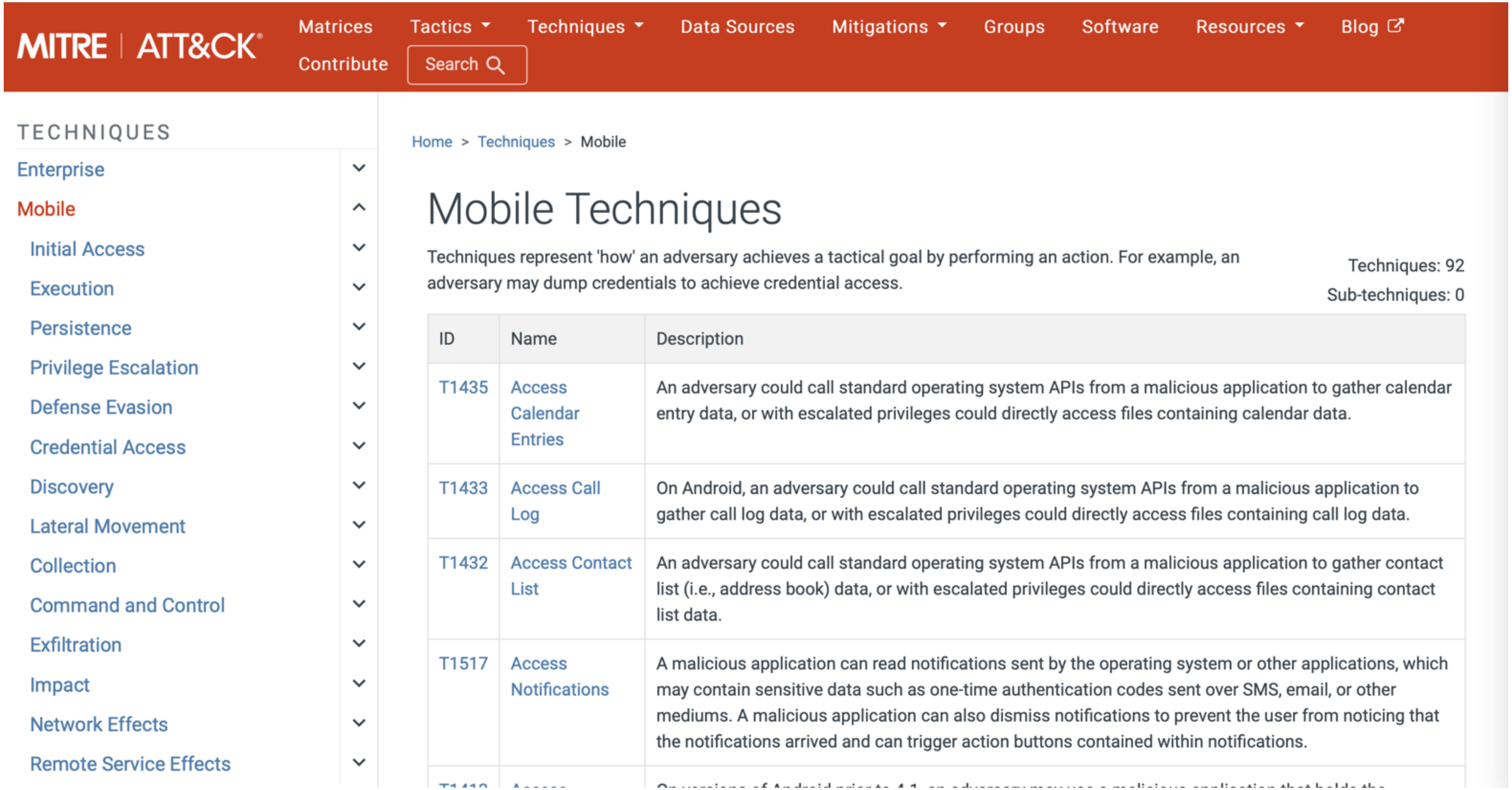
Attack trees


Payment information is stolen by attackers
Attacker sniffs payment information as it is being submitted to the database
Attacker has access to unencrypted payment database
Attacker gains trusted access to payment database
Attacker moves laterally to server containing payment database
Attacker has physical access to database server
Attacker has access to unencrypted payment database backups
Attacker has a network connection to the payment database service
Attacker has database administrative credentials
Attacker has access to encrypted payment database
Attacker has access to decryption keys

Bottom-up approach: starts from the damage

ATT&CK framework

ATT&CK is a public repository and framework for capturing and describing what attackers have done based on real-world data (https://attack.mitre.org)



Mobile Techniques

Techniques represent 'how' an adversary achieves a tactical goal by performing an action. For example, an adversary may dump credentials to achieve credential access.

Techniques: 92

Sub-techniques: 0

ID	Name	Description
T1435	Access Calendar Entries	An adversary could call standard operating system APIs from a malicious application to gather calendar entry data, or with escalated privileges could directly access files containing calendar data.
T1433	Access Call Log	On Android, an adversary could call standard operating system APIs from a malicious application to gather call log data, or with escalated privileges could directly access files containing call log data.
T1432	Access Contact List	An adversary could call standard operating system APIs from a malicious application to gather contact list (i.e., address book) data, or with escalated privileges could directly access files containing contact list data.
T1517	Access Notifications	A malicious application can read notifications sent by the operating system or other applications, which may contain sensitive data such as one-time authentication codes sent over SMS, email, or other mediums. A malicious application can also dismiss notifications to prevent the user from noticing that the notifications arrived and can trigger action buttons contained within notifications.
T1413	Access	On versions of Android prior to 4.1, an adversary may use a malicious application that holds the

Scoring

IEEE Std 11073-40101-2020
Health informatics—Device interoperability
Part 40101: Foundational—Cybersecurity—Processes for vulnerability assessment

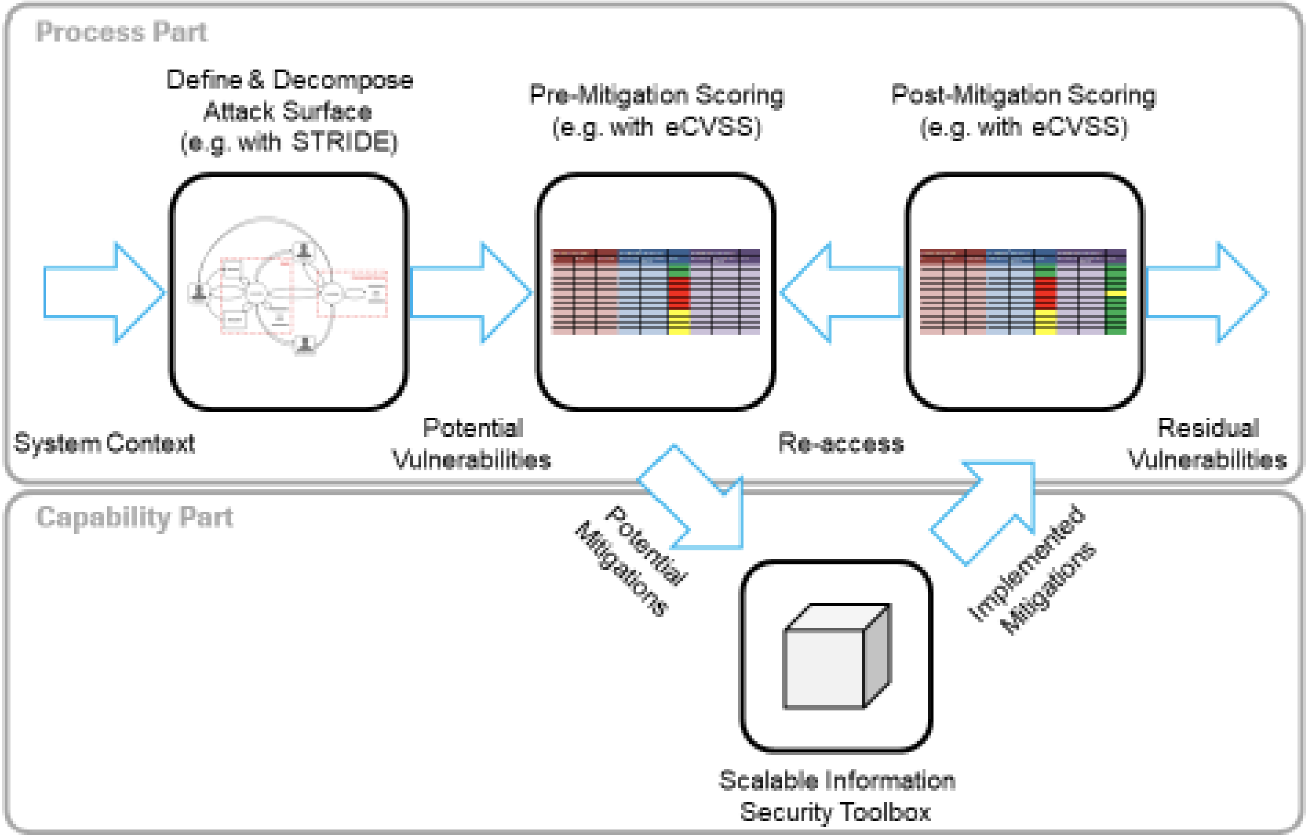
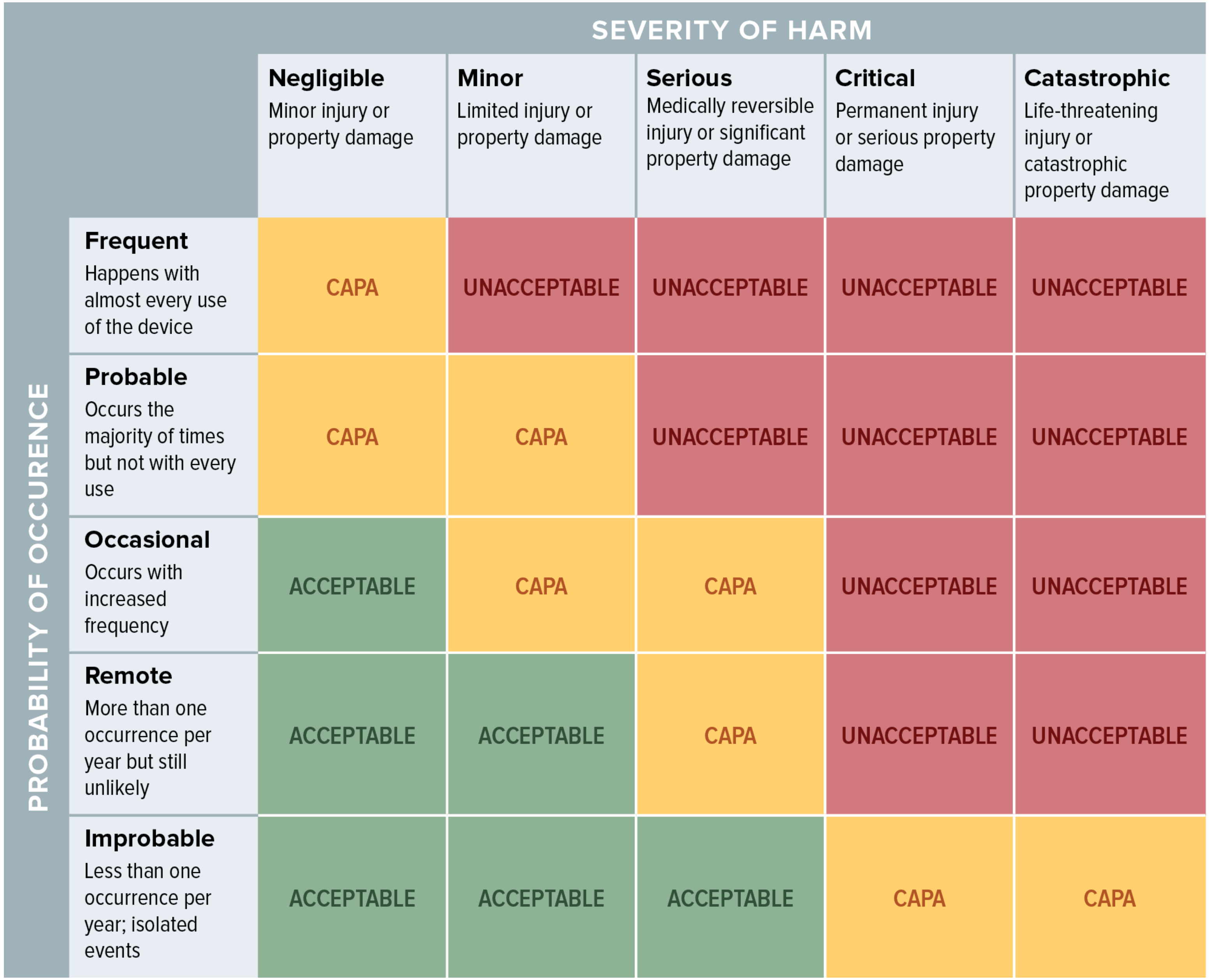

Process Part
Define & Decompose Attack Surface (e.g. with STRIDE)
Pre-Mitigation Scoring (e.g. with eCVSS)
Post-Mitigation Scoring (e.g. with eCVSS)
System Context
Potential Vulnerabilities
Re-access
Residual Vulnerabilities
Capability Part
Potential Mitigations
Implemented Mitigations
Scalable Information Security Toolbox

Figure 1—Vulnerability assessment workflow

Assessment of impact and likelihood

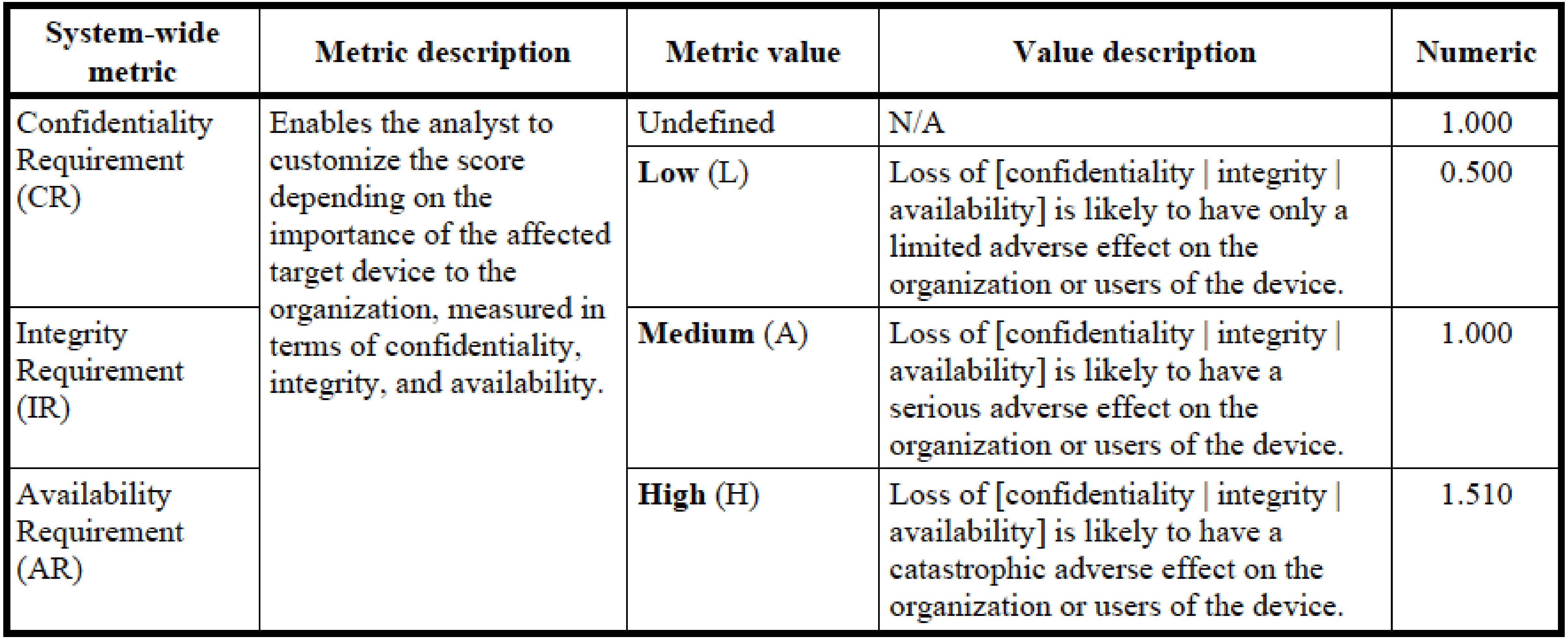


		SEVERITY OF HARM				
		Negligible Minor injury or property damage	**Minor** Limited injury or property damage	**Serious** Medically reversible injury or significant property damage	**Critical** Permanent injury or serious property damage	**Catastrophic** Life-threatening injury or catastrophic property damage
PROBABILITY OF OCCURENCE	**Frequent** Happens with almost every use of the device	CAPA	UNACCEPTABLE	UNACCEPTABLE	UNACCEPTABLE	UNACCEPTABLE
	Probable Occurs the majority of times but not with every use	CAPA	CAPA	UNACCEPTABLE	UNACCEPTABLE	UNACCEPTABLE
	Occasional Occurs with increased frequency	ACCEPTABLE	CAPA	CAPA	UNACCEPTABLE	UNACCEPTABLE
	Remote More than one occurrence per year but still unlikely	ACCEPTABLE	ACCEPTABLE	CAPA	UNACCEPTABLE	UNACCEPTABLE
	Improbable Less than one occurrence per year; isolated events	ACCEPTABLE	ACCEPTABLE	ACCEPTABLE	CAPA	CAPA

embedded Common vulnerability Scoring System –eCVSS

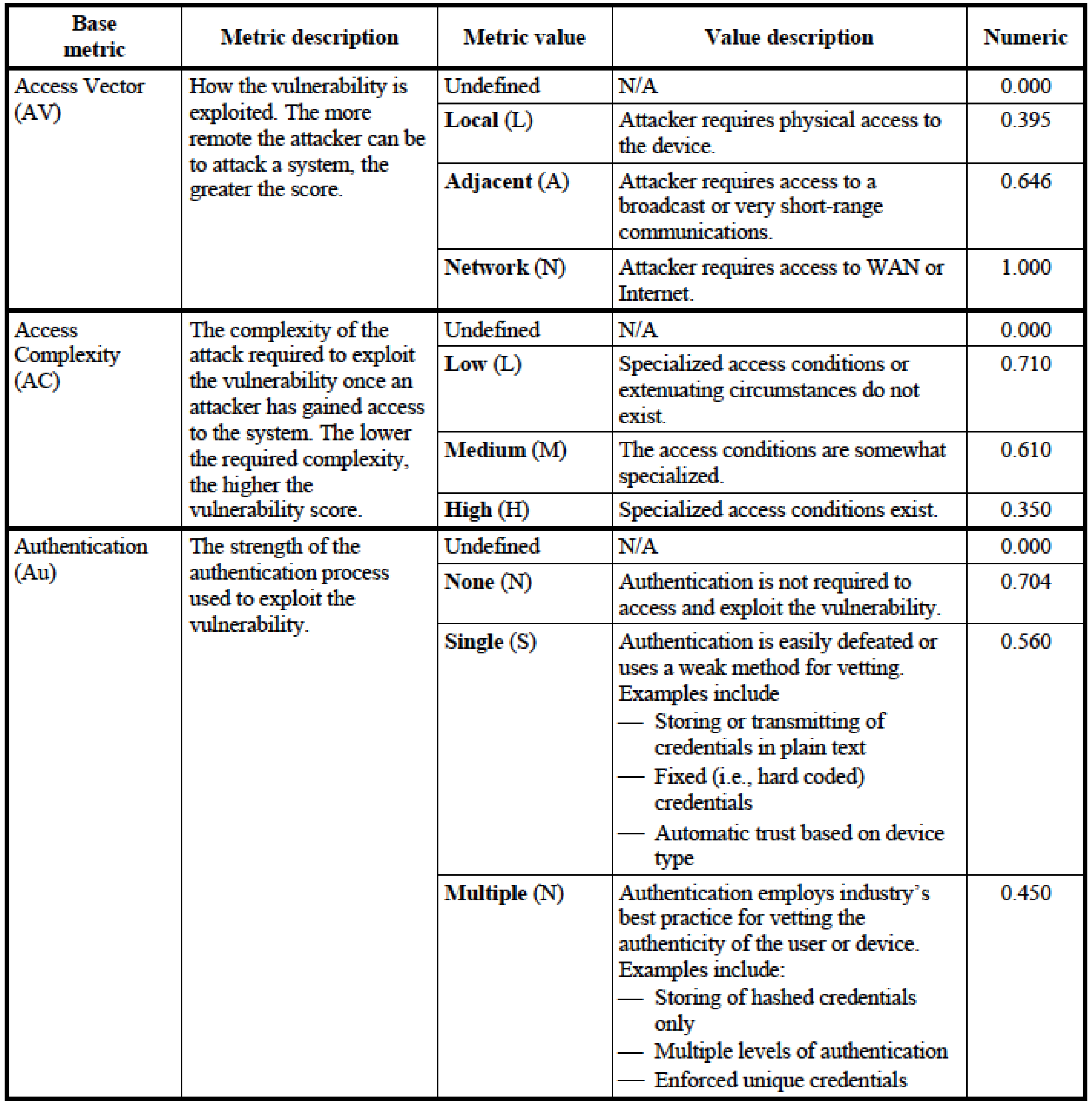
- The assessment has to be done before and after mitigation measures are put in place
- Even in the case of security/privacy by design the mitigation measures are not considered in the first assessment
- Two-level assessment:
  - System-wide level (system-wide metric): Represents the system requirements for a confidentiality, integrity, and availability (CIA) triad that are set once for the product and then applied to all vulnerabilities
  - Vulnerability level
    - base metric: intrinsic and fundamental characteristics of a vulnerability that are constant over time and user environments.
    - environmental metric: the characteristics of a vulnerability that are relevant and unique to a particular user’s environment.

eCVSS system wide metric



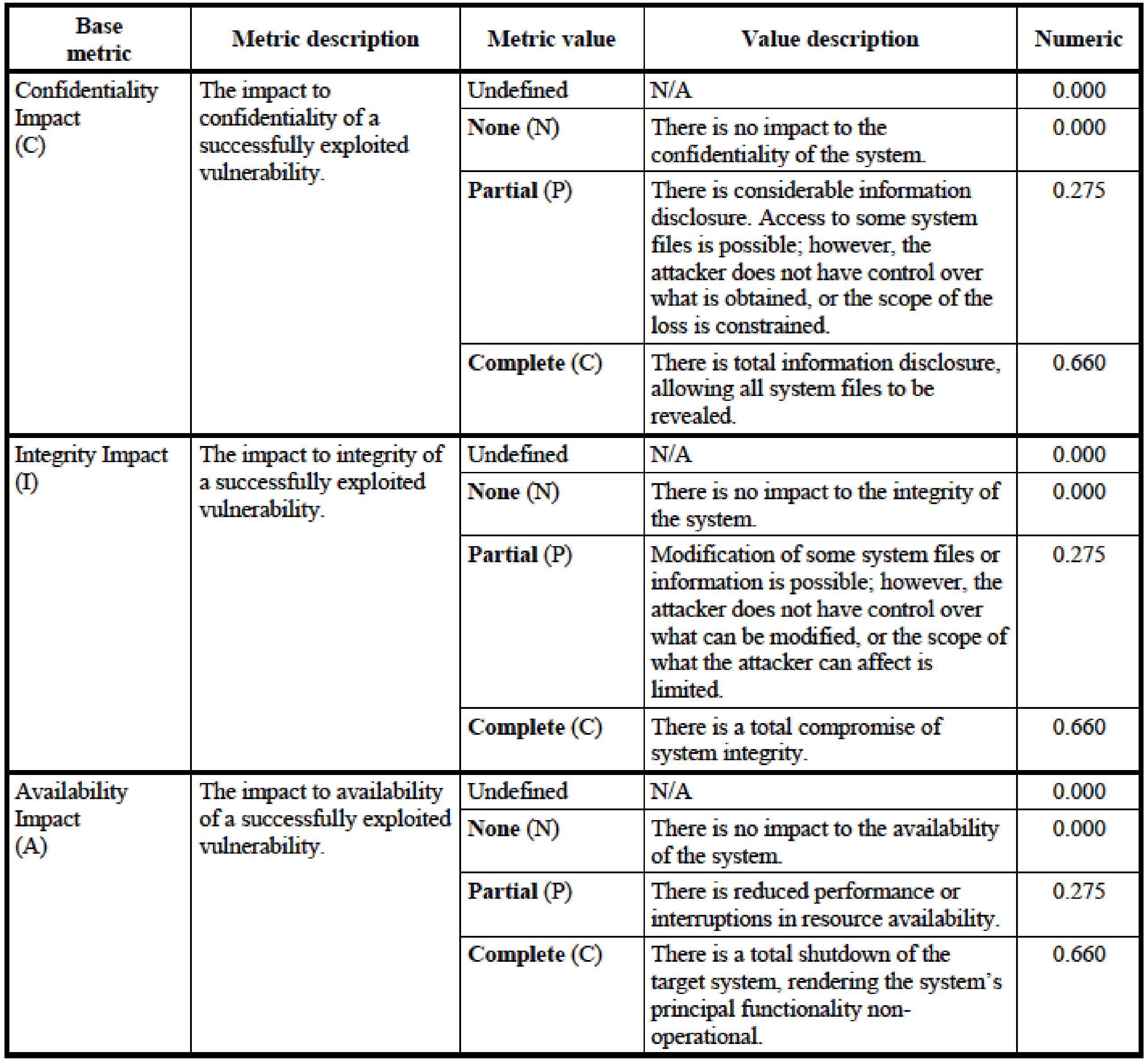
System-wide metric	**Metric description**	**Metric value**	**Value description**	**Numeric**
Confidentiality Requirement (CR)	Enables the analyst to customize the score depending on the importance of the affected target device to the organization, measured in terms of confidentiality, integrity, and availability.	Undefined	N/A	1.000
		Low (L)	Loss of [confidentiality | integrity | availability] is likely to have only a limited adverse effect on the organization or users of the device.	0.500
Integrity Requirement (IR)		**Medium** (A)	Loss of [confidentiality | integrity | availability] is likely to have a serious adverse effect on the organization or users of the device.	1.000
Availability Requirement (AR)		**High** (H)	Loss of [confidentiality | integrity | availability] is likely to have a catastrophic adverse effect on the organization or users of the device.	1.510

eCVSS Base metric 1



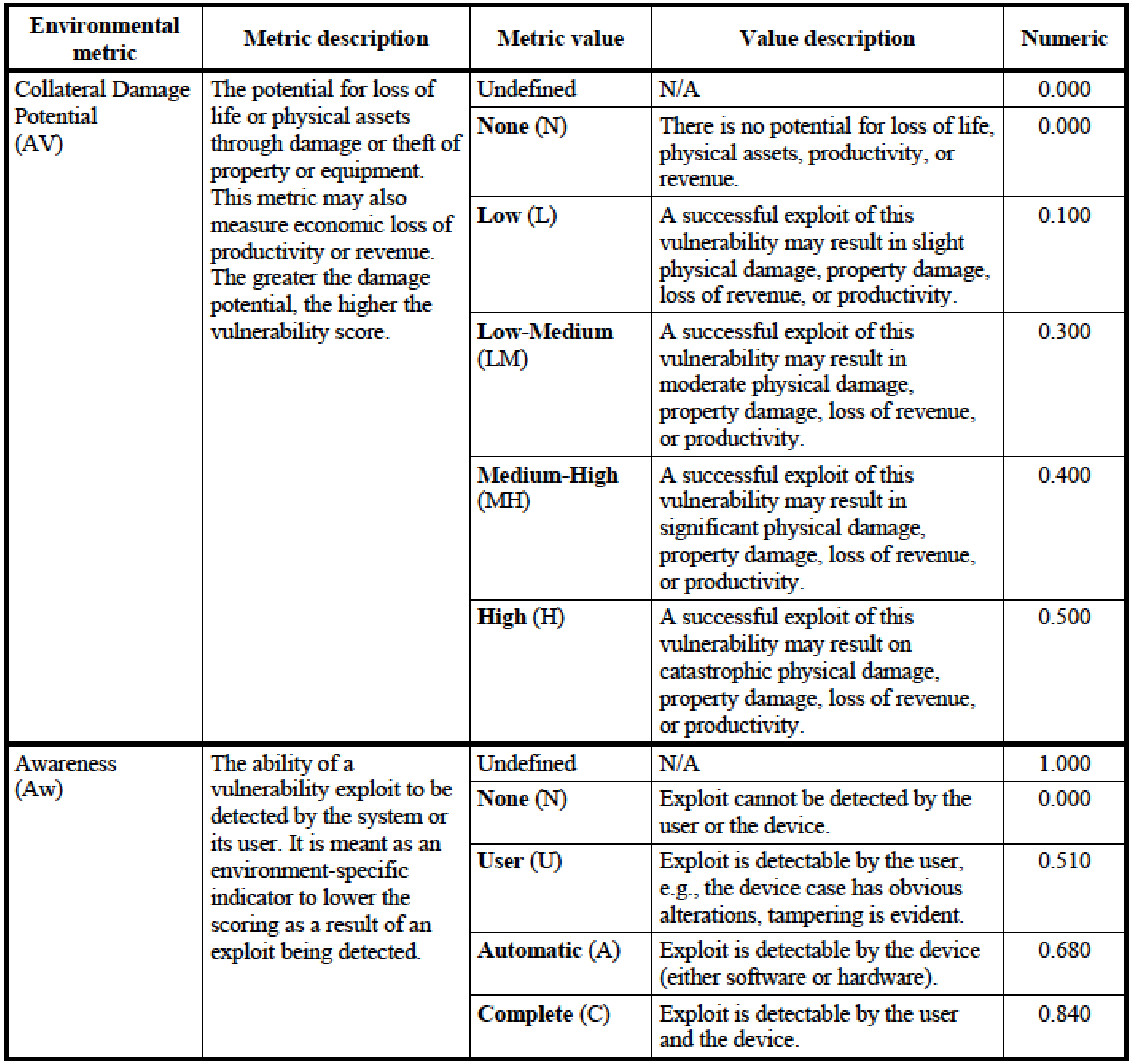
Base metric	Metric description	Metric value	Value description	Numeric
Access Vector (AV)	How the vulnerability is exploited. The more remote the attacker can be to attack a system, the greater the score.	Undefined	N/A	0.000
		Local (L)	Attacker requires physical access to the device.	0.395
		Adjacent (A)	Attacker requires access to a broadcast or very short-range communications.	0.646
		Network (N)	Attacker requires access to WAN or Internet.	1.000
Access Complexity (AC)	The complexity of the attack required to exploit the vulnerability once an attacker has gained access to the system. The lower the required complexity, the higher the vulnerability score.	Undefined	N/A	0.000
		Low (L)	Specialized access conditions or extenuating circumstances do not exist.	0.710
		Medium (M)	The access conditions are somewhat specialized.	0.610
		High (H)	Specialized access conditions exist.	0.350
Authentication (Au)	The strength of the authentication process used to exploit the vulnerability.	Undefined	N/A	0.000
		None (N)	Authentication is not required to access and exploit the vulnerability.	0.704
		Single (S)	Authentication is easily defeated or uses a weak method for vetting. Examples include — Storing or transmitting of credentials in plain text — Fixed (i.e., hard coded) credentials — Automatic trust based on device type	0.560
		Multiple (N)	Authentication employs industry's best practice for vetting the authenticity of the user or device. Examples include: — Storing of hashed credentials only — Multiple levels of authentication — Enforced unique credentials	0.450

eCVSS Base metric 2



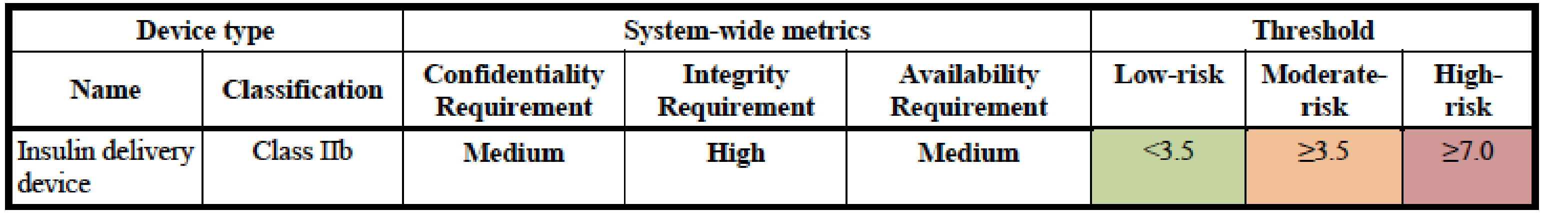
Base metric	Metric description	Metric value	Value description	Numeric
Confidentiality Impact (C)	The impact to confidentiality of a successfully exploited vulnerability.	Undefined	N/A	0.000
		None (N)	There is no impact to the confidentiality of the system.	0.000
		Partial (P)	There is considerable information disclosure. Access to some system files is possible; however, the attacker does not have control over what is obtained, or the scope of the loss is constrained.	0.275
		Complete (C)	There is total information disclosure, allowing all system files to be revealed.	0.660
Integrity Impact (I)	The impact to integrity of a successfully exploited vulnerability.	Undefined	N/A	0.000
		None (N)	There is no impact to the integrity of the system.	0.000
		Partial (P)	Modification of some system files or information is possible; however, the attacker does not have control over what can be modified, or the scope of what the attacker can affect is limited.	0.275
		Complete (C)	There is a total compromise of system integrity.	0.660
Availability Impact (A)	The impact to availability of a successfully exploited vulnerability.	Undefined	N/A	0.000
		None (N)	There is no impact to the availability of the system.	0.000
		Partial (P)	There is reduced performance or interruptions in resource availability.	0.275
		Complete (C)	There is a total shutdown of the target system, rendering the system's principal functionality non-operational.	0.660

eCVSS Environmental metric

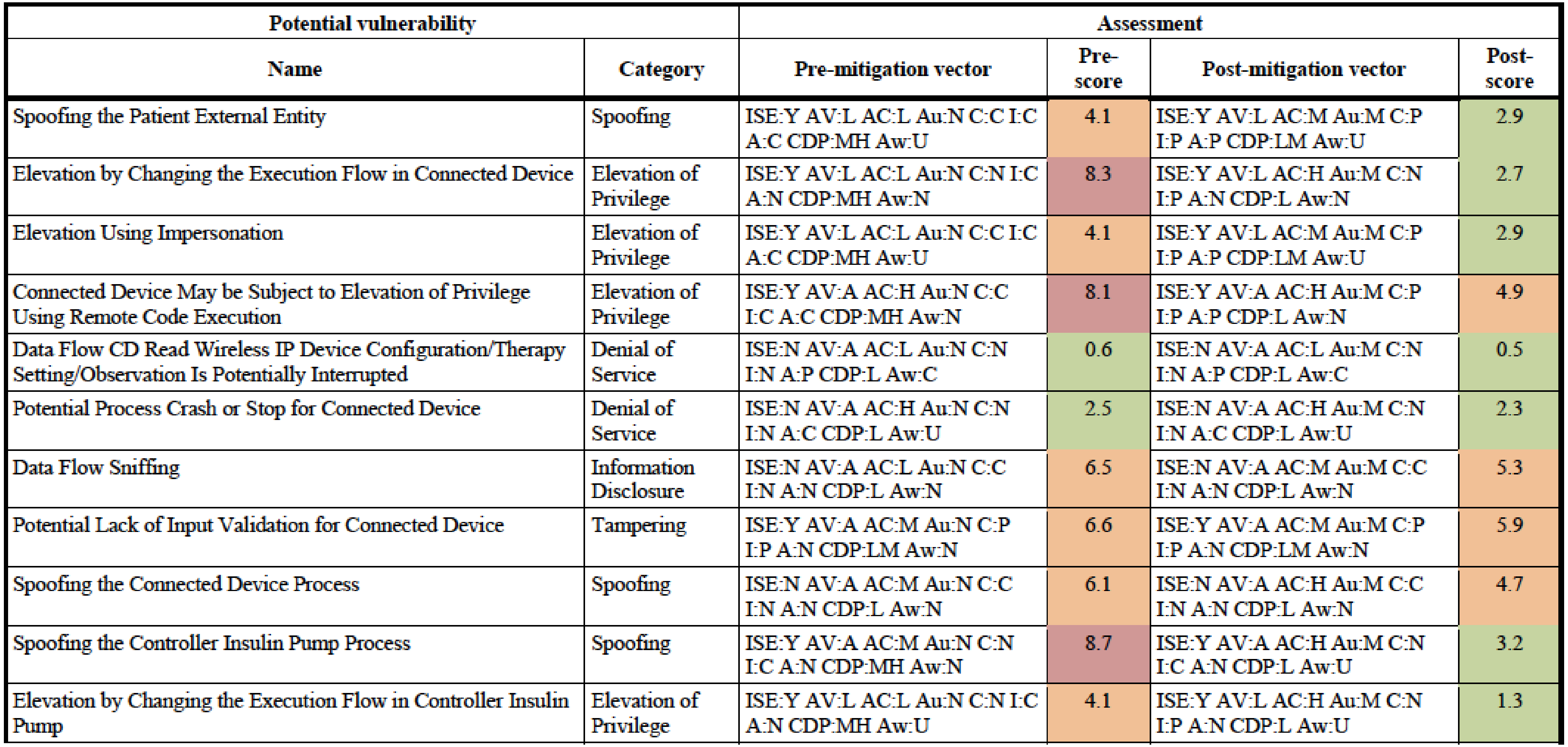


Environmental metric	Metric description	Metric value	Value description	Numeric
Collateral Damage Potential (AV)	The potential for loss of life or physical assets through damage or theft of property or equipment. This metric may also measure economic loss of productivity or revenue. The greater the damage potential, the higher the vulnerability score.	Undefined	N/A	0.000
		None (N)	There is no potential for loss of life, physical assets, productivity, or revenue.	0.000
		Low (L)	A successful exploit of this vulnerability may result in slight physical damage, property damage, loss of revenue, or productivity.	0.100
		Low-Medium (LM)	A successful exploit of this vulnerability may result in moderate physical damage, property damage, loss of revenue, or productivity.	0.300
		Medium-High (MH)	A successful exploit of this vulnerability may result in significant physical damage, property damage, loss of revenue, or productivity.	0.400
		High (H)	A successful exploit of this vulnerability may result on catastrophic physical damage, property damage, loss of revenue, or productivity.	0.500
Awareness (Aw)	The ability of a vulnerability exploit to be detected by the system or its user. It is meant as an environment-specific indicator to lower the scoring as a result of an exploit being detected.	Undefined	N/A	1.000
		None (N)	Exploit cannot be detected by the user or the device.	0.000
		User (U)	Exploit is detectable by the user, e.g., the device case has obvious alterations, tampering is evident.	0.510
		Automatic (A)	Exploit is detectable by the device (either software or hardware).	0.680
		Complete (C)	Exploit is detectable by the user and the device.	0.840

eCVSS example



Device type		System-wide metrics			Threshold		
Name	Classification	Confidentiality Requirement	Integrity Requirement	Availability Requirement	Low-risk	Moderate-risk	High-risk
Insulin delivery device	Class IIb	**Medium**	**High**	**Medium**	<3.5	≥3.5	≥7.0



Potential vulnerability		Assessment			
Name	Category	Pre-mitigation vector	Pre-score	Post-mitigation vector	Post-score
Spoofing the Patient External Entity	Spoofing	ISE:Y AV:L AC:L Au:N C:C I:C A:C CDP:MH Aw:U	4.1	ISE:Y AV:L AC:M Au:M C:P I:P A:P CDP:LM Aw:U	2.9
Elevation by Changing the Execution Flow in Connected Device	Elevation of Privilege	ISE:Y AV:L AC:L Au:N C:N I:C A:N CDP:MH Aw:N	8.3	ISE:Y AV:L AC:H Au:M C:N I:P A:N CDP:L Aw:N	2.7
Elevation Using Impersonation	Elevation of Privilege	ISE:Y AV:L AC:L Au:N C:C I:C A:C CDP:MH Aw:U	4.1	ISE:Y AV:L AC:M Au:M C:P I:P A:P CDP:LM Aw:U	2.9
Connected Device May be Subject to Elevation of Privilege Using Remote Code Execution	Elevation of Privilege	ISE:Y AV:A AC:H Au:N C:C I:C A:C CDP:MH Aw:N	8.1	ISE:Y AV:A AC:H Au:M C:P I:P A:P CDP:L Aw:N	4.9
Data Flow CD Read Wireless IP Device Configuration/Therapy Setting/Observation Is Potentially Interrupted	Denial of Service	ISE:N AV:A AC:L Au:N C:N I:N A:P CDP:L Aw:C	0.6	ISE:N AV:A AC:L Au:M C:N I:N A:P CDP:L Aw:C	0.5
Potential Process Crash or Stop for Connected Device	Denial of Service	ISE:N AV:A AC:H Au:N C:N I:N A:C CDP:L Aw:U	2.5	ISE:N AV:A AC:H Au:M C:N I:N A:C CDP:L Aw:U	2.3
Data Flow Sniffing	Information Disclosure	ISE:N AV:A AC:L Au:N C:C I:N A:N CDP:L Aw:N	6.5	ISE:N AV:A AC:M Au:M C:C I:N A:N CDP:L Aw:N	5.3
Potential Lack of Input Validation for Connected Device	Tampering	ISE:Y AV:A AC:M Au:N C:P I:P A:N CDP:LM Aw:N	6.6	ISE:Y AV:A AC:M Au:M C:P I:P A:N CDP:LM Aw:N	5.9
Spoofing the Connected Device Process	Spoofing	ISE:N AV:A AC:M Au:N C:C I:N A:N CDP:L Aw:N	6.1	ISE:N AV:A AC:H Au:M C:C I:N A:N CDP:L Aw:N	4.7
Spoofing the Controller Insulin Pump Process	Spoofing	ISE:Y AV:A AC:M Au:N C:N I:C A:N CDP:MH Aw:N	8.7	ISE:Y AV:A AC:H Au:M C:N I:C A:N CDP:L Aw:U	3.2
Elevation by Changing the Execution Flow in Controller Insulin Pump	Elevation of Privilege	ISE:Y AV:L AC:L Au:N C:N I:C A:N CDP:MH Aw:U	4.1	ISE:Y AV:L AC:H Au:M C:N I:P A:N CDP:L Aw:U	1.3

Mitigation measures

- There are four main strategies for addressing threats:
  - Eliminate
  - **Mitigate**
  - Accept
  - Transfer
- Major mitigation measures:
  - Protect
  - Detect
  - Respond
  - Recover

Mitigation measures

IEEE 11073 - 40102

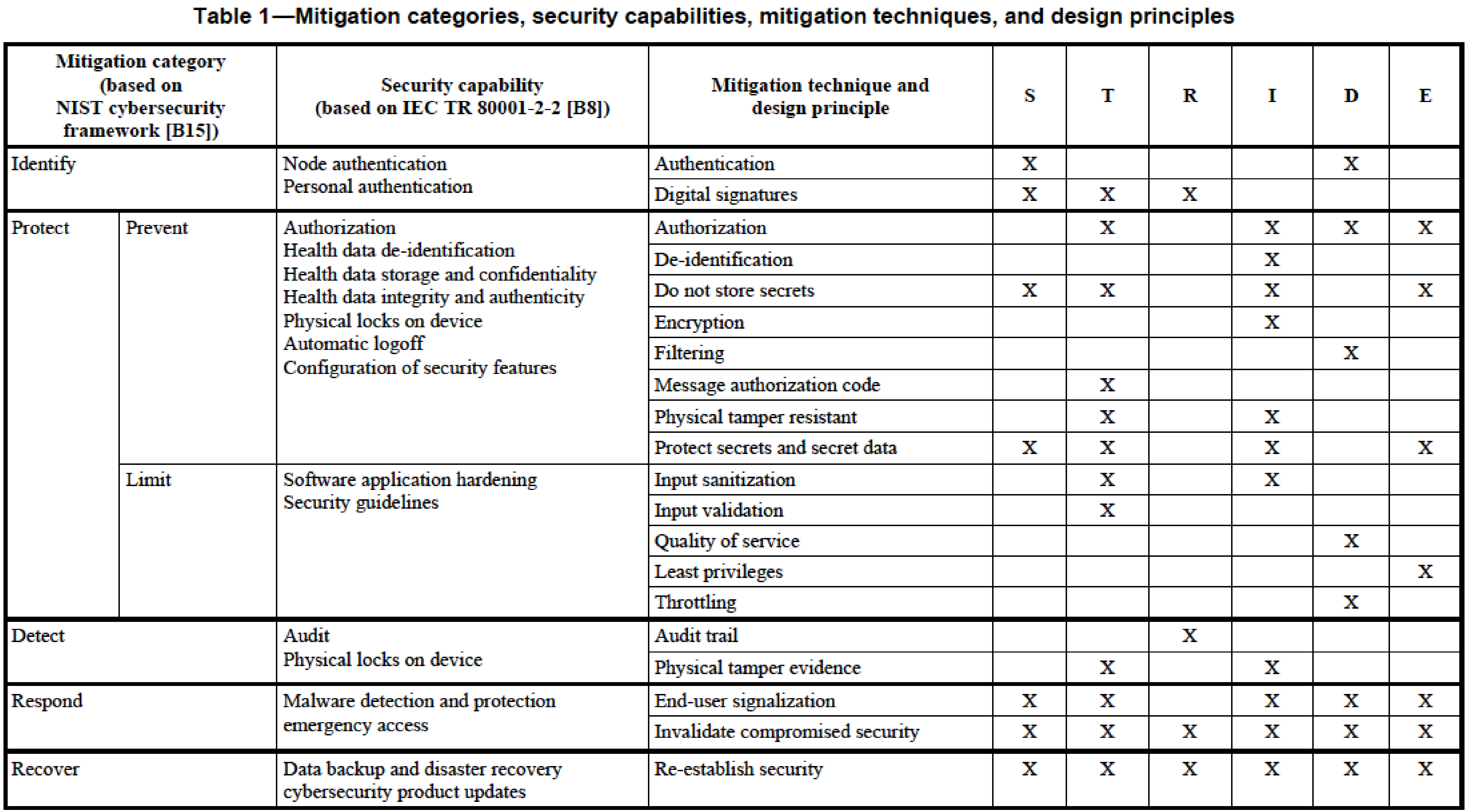
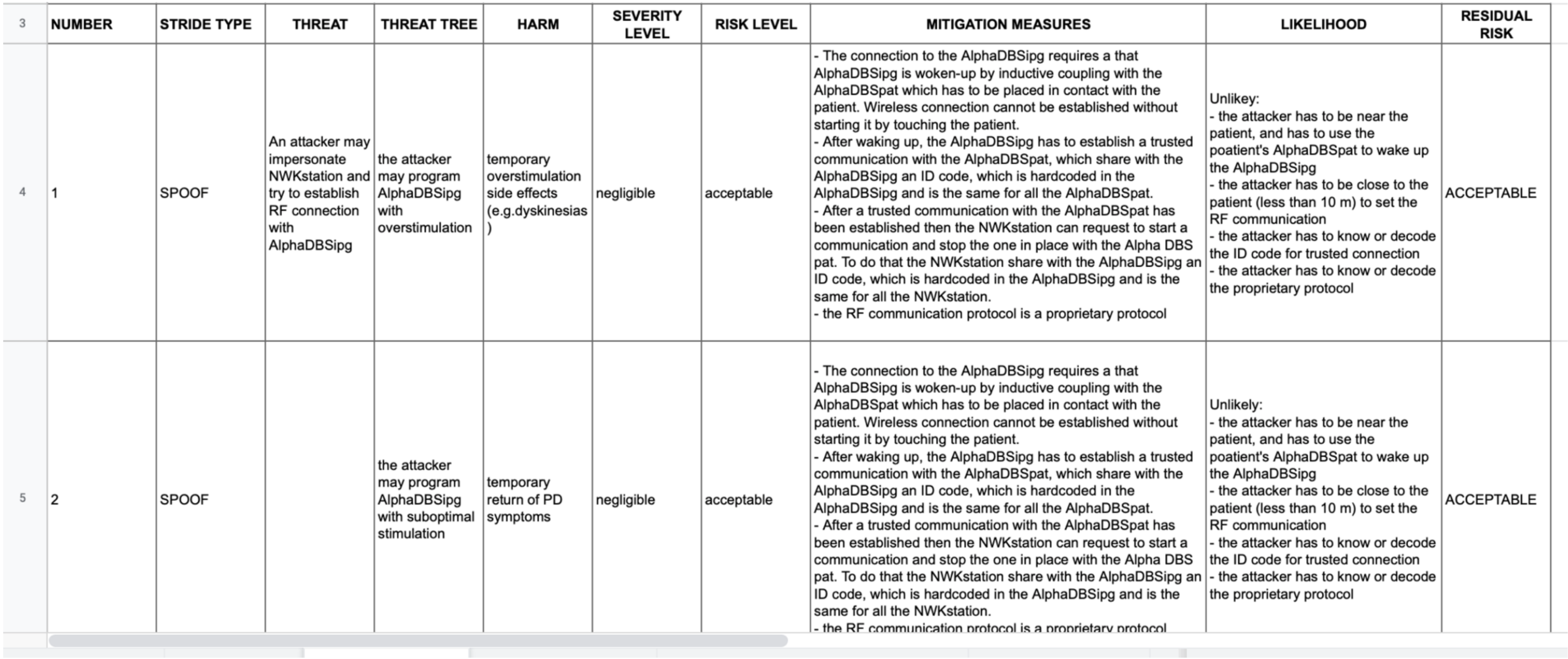


Table 1—Mitigation categories, security capabilities, mitigation techniques, and design principles

Mitigation category (based on NIST cybersecurity framework [B15])		Security capability (based on IEC TR 80001-2-2 [B8])	Mitigation technique and design principle	S	T	R	I	D	E
Identify		Node authentication Personal authentication	Authentication	X				X	
			Digital signatures	X	X	X			
Protect	Prevent	Authorization Health data de-identification Health data storage and confidentiality Health data integrity and authenticity Physical locks on device Automatic logoff Configuration of security features	Authorization		X		X	X	X
			De-identification				X		
			Do not store secrets	X	X		X		X
			Encryption				X		
			Filtering					X	
			Message authorization code		X				
			Physical tamper resistant		X		X		
			Protect secrets and secret data	X	X		X		X
	Limit	Software application hardening Security guidelines	Input sanitization		X		X		
			Input validation		X				
			Quality of service					X	
			Least privileges						X
			Throttling					X	
Detect		Audit Physical locks on device	Audit trail			X			
			Physical tamper evidence		X		X		
Respond		Malware detection and protection emergency access	End-user signalization	X	X		X	X	X
			Invalidate compromised security	X	X	X	X	X	X
Recover		Data backup and disaster recovery cybersecurity product updates	Re-establish security	X	X	X	X	X	X

Cybersecurity risk assessment matrix



3	NUMBER	STRIDE TYPE	THREAT	THREAT TREE	HARM	SEVERITY LEVEL	RISK LEVEL	MITIGATION MEASURES	LIKELIHOOD	RESIDUAL RISK
4	1	SPOOF	An attacker may impersonate NWKstation and try to establish RF connection with AlphaDBSipg	the attacker may program AlphaDBSipg with overstimulation	temporary overstimulation side effects (e.g.dyskinesias)	negligible	acceptable	- The connection to the AlphaDBSipg requires a that AlphaDBSipg is woken-up by inductive coupling with the AlphaDBSpat which has to be placed in contact with the patient. Wireless connection cannot be established without starting it by touching the patient. - After waking up, the AlphaDBSipg has to establish a trusted communication with the AlphaDBSpat, which share with the AlphaDBSipg an ID code, which is hardcoded in the AlphaDBSipg and is the same for all the AlphaDBSpat. - After a trusted communication with the AlphaDBSpat has been established then the NWKstation can request to start a communication and stop the one in place with the Alpha DBS pat. To do that the NWKstation share with the AlphaDBSipg an ID code, which is hardcoded in the AlphaDBSipg and is the same for all the NWKstation. - the RF communication protocol is a proprietary protocol	Unlikey: - the attacker has to be near the patient, and has to use the poatient's AlphaDBSpat to wake up the AlphaDBSipg - the attacker has to be close to the patient (less than 10 m) to set the RF communication - the attacker has to know or decode the ID code for trusted connection - the attacker has to know or decode the proprietary protocol	ACCEPTABLE
5	2	SPOOF		the attacker may program AlphaDBSipg with suboptimal stimulation	temporary return of PD symptoms	negligible	acceptable	- The connection to the AlphaDBSipg requires a that AlphaDBSipg is woken-up by inductive coupling with the AlphaDBSpat which has to be placed in contact with the patient. Wireless connection cannot be established without starting it by touching the patient. - After waking up, the AlphaDBSipg has to establish a trusted communication with the AlphaDBSpat, which share with the AlphaDBSipg an ID code, which is hardcoded in the AlphaDBSipg and is the same for all the AlphaDBSpat. - After a trusted communication with the AlphaDBSpat has been established then the NWKstation can request to start a communication and stop the one in place with the Alpha DBS pat. To do that the NWKstation share with the AlphaDBSipg an ID code, which is hardcoded in the AlphaDBSipg and is the same for all the NWKstation. - the RF communication protocol is a proprietary protocol	Unlikely: - the attacker has to be near the patient, and has to use the poatient's AlphaDBSpat to wake up the AlphaDBSipg - the attacker has to be close to the patient (less than 10 m) to set the RF communication - the attacker has to know or decode the ID code for trusted connection - the attacker has to know or decode the proprietary protocol	ACCEPTABLE

Traceability matrix

Traceability among requirements, specifications, identified hazards and mitigations, and Verification and Validation testing.

Mitigation measure	System requirement	Test case	Test execution	Test result	Issues

REAL CASE

CISA issues Security Alert for Customers Affected by Oracle Data Breach

Posted By Steve Alder on Apr 21, 2025

The U.S. Cybersecurity and Infrastructure Security Agency (CISA) has issued a security alert about the recently confirmed Oracle data breach. Oracle has confirmed that an unauthorized individual gained access to its legacy cloud environment, although limited details about the incident have been disclosed by Oracle, and the extent of the breach is currently unconfirmed. There have been reports of threat actor activity targeting Oracle customers, but the scope and impact of that activity are not yet known.

Information compromised in the incident includes **credentials such as usernames, email addresses, passwords, authentication tokens, and encryption keys**, and as such, the breach poses a risk to enterprise environments. CISA recommends that Oracle customers take steps to protect against **unauthorized access** and warns that when credential material has been **embedded into scripts**, applications, infrastructure templates, and automation tools, it can be hard to detect. Should action not be taken, unauthorized actors could potentially use credential material for long-term access to enterprise environments.

Breaches of credential material carry a risk, as **threat actors frequently harvest and weaponize credentials**. The stolen data can be enriched with information obtained in prior breaches, the information could be sold to other threat actors, and could be used to conduct BEC attacks or phishing campaigns. **Valid credentials could be used to escalate privileges and move laterally within networks, or access cloud and identity management systems**.

https://www.hipaajournal.com/oracle-health-data-breach/

TIMELINE

March 31, 2025: Oracle Health Breach Affects Patients of Multiple U.S. Hospitals

Oracle suffered **two security incidents**. The first involved Oracle Health (formerly Cerner), where a cyberattack on a legacy server allowed a threat actor to **exfiltrate electronic health record (EHR) data using stolen credentials**. Oracle Health detected the breach on February 20, 2025, with the breach dating back to at least January 22, 2025. Affected healthcare providers are being notified but must handle HIPAA breach notifications themselves. Extortion attempts by a threat actor named "Andrew" have been reported.

The second incident involved an individual (rose87168) claiming to **exploit a vulnerability (CVE-2021-35587)** in **Oracle Access Manager**, allegedly stealing about **6 million records** containing sensitive authentication data. Oracle Cloud denies any breach but has not offered a full explanation, despite confirmations from affected companies that the leaked data is genuine. (https://nvd.nist.gov/vuln/detail/cve-2021-35587)

April 3, 2025: Oracle Sued Over Healthcare Data Breach

A class action lawsuit was filed against Oracle Corporation in the U.S. District Court for the Western District of Texas by Michael Toikach, after a January 2025 data breach. Oracle has not publicly confirmed the breach yet, **and it is not listed on the HHS OCR breach portal**. The lawsuit claims that Oracle failed **to use reasonable security measures** to protect personal and health information stored through a healthcare provider using Oracle software. Alleged failures include poor network segmentation, insufficient cybersecurity training, and lack of monitoring systems. The breach was detected on February 20, 2025, but plaintiffs argue Oracle delayed required breach notifications under HIPAA and Texas law. Plaintiffs say the delay and lack of transparency put them at greater risk of identity theft and fraud. The lawsuit seeks compensatory damages, reimbursement, long-term credit monitoring, and injunctive relief demanding major security improvements like encryption, penetration testing, audits, and better security training.

April 15, 2025: Oracle Confirms Hacking Incident Involving Obsolete Servers

Oracle notified customers of a security incident but confirmed **that Oracle Cloud Infrastructure (OCI) was not breached**. Instead, **a hacker accessed two obsolete servers (not part of OCI), exposing usernames but no usable passwords or customer data**. Security researcher Kevin Beaumont criticized Oracle for downplaying the breach, noting **the compromised servers were Oracle-managed cloud services (Oracle Cloud Classic**). **Separately, Oracle Health (formerly Cerner)** suffered a breach **involving legacy servers** not yet migrated to Oracle Cloud, with **stolen credentials used to access them**. A hacker is allegedly trying to **extort Oracle Health customers by threatening to release stolen data**. A lawsuit has been filed, accusing Oracle Health of negligence after sensitive information like Social Security numbers and clinical results were stolen. Plaintiffs claim they were not properly notified and now face risks of identity theft. Oracle Health stated that healthcare providers must assess and handle HIPAA breach notifications themselves.

MITIGATION

The suggested mitigations include **resetting passwords across enterprise servers**, especially in cases where **local credentials may not be federated through enterprise identity solutions**. Source code should be reviewed, along with infrastructure as code templates, configuration files, and automation templates, **to identify embedded credentials**, which should be replaced with secure authentication methods. **Authentication logs** should be **monitored** for anomalous activity, especially for privileged, service, or federated identity accounts, and **if possible, phishing-resistant multifactor authentication should be implemented and enforced**, especially for administrator accounts.

Oracle has stressed that the breach involved **legacy servers** and there was no breach of Oracle Cloud, but has yet to issue any public advisory to help customers mitigate risk.