

Common Cybersecurity Requirements in IoT Standards, Best Practices, and Guidelines

Rauli Kaksonen^a, Kimmo Halunen^b and Juha Rönning^c

University of Oulu, Oulu, Finland

Keywords: Internet of Things, IoT, Cybersecurity, Security Requirements, Standards, Best Practices, Guidelines.

Abstract: The cybersecurity of the Internet of Things (IoT) is an increasing concern and product vendors are advised to follow security standards, best practices, and guidelines. From the many requirement sources, a vendor is likely to choose only a few. How does this selection impact the security requirements of an IoT product? To answer the question, we collect requirements from 16 sources and divide them into categories for comparison. Common categories are identified, with all sources covering Security design, Interface security, Authentication, Data protection, and System updates. The agreement on the high-level categories does not hold in the subcategories and the selection of the sources have a big impact to the requirement details. Consolidation of the IoT security requirements would be desirable and possible.

1 INTRODUCTION

As more devices are connected to the Internet of Things (IoT), their cybersecurity is an increasing concern. IoT has a great impact on the safety, security, and privacy of people (ENISA, 2017; NIST, 2018). There are many cases of IoT security breaches and the vendors do not appear to have security as a key requirement (Sayegh, 2021). Momenzadeh et al. examined two IoT devices and concluded that would they have followed some well-known security best practices, they would be more secure (Momenzadeh et al., 2020). Stellios et al. performed a survey of IoT cyberattacks and noted that the majority of attacks could be mitigated if the existing security mechanisms and standards would have been properly implemented (Stellios et al., 2018).

In addition to countless Internet sites, there are many sources of IoT security requirements, such as academic studies (Momenzadeh et al., 2020; Stellios et al., 2018; Tange et al., 2020). System builders and administrators have their sources, e.g. from NIST (NIST, 2018). Organizations like ENISA, NIST, and many others have published requirements for IoT product security (ENISA, 2017; NIST, 2020). For practical reasons, IoT vendors are likely to choose


just a few sources for requirements. How does this selection impact the security requirements for an IoT product? How much work it is to cover additional requirement sources? This depends on how much the requirements differ between the sources, but this does not appear to be studied. To fill the gap, this study aims to answer the following research questions:


1. Is there a consensus on security requirements in security standards, guidelines, best practices, and other sources?
2. What are the common security requirements?
3. How well do the sources cover the common security requirements?
4. How much do the security requirements differ between the sources?


2 METHODS

2.1 Security Requirements and Categories

To answer the research questions, we need to analyze IoT security requirement sources, such as standards, guidelines, and best practices. For a source to be included it has to cover security broadly, e.g. not only privacy or data protection. The source has to be authored by a respected organization, such as a govern-

^a  <https://orcid.org/0000-0001-8692-763X>

^b  <https://orcid.org/0000-0003-1169-5920>

^c  <https://orcid.org/0000-0001-9993-8602>

ment body or industry consortium, and available without a fee. The source must be from the year 2016 or newer. Sources use various terms, e.g. security principles, controls, capabilities, but security *requirement* is the most common term and used in this paper. The style in the sources vary, e.g. consider the following requirements.

“Apply authentication functions for each IoT system or service that will ensure the security of the entire IoT system or service” (IoTAC, 2016)

“Confirm that the device supports user authentication to be able to make changes to the device configuration with the goal to require authentication before changes are made, reducing risk to the device that anyone can walk up and make anonymous changes to the device.” (CTIA, 2021)

“Authentication mechanisms used to authenticate users against a device shall use best practice cryptography, appropriate to the properties of the technology, risk and usage.” (ETSI, 2020)

The differences make it hard to determine when sources share the same requirement. As a solution, requirements are divided into *categories*, such as *cryptography* and *authentication*, which can be compared across sources. Complex requirements are split into *individual requirements* each matching a single category. In the rest of this article, the term requirement generally means the individual requirement.

The categories in requirements can be generic or specific, e.g. *passwords* is more specific than *authentication*. To capture this, the categories are arranged into a taxonomy where the specific categories are linked to the generic ones. A category can be specific for several generic categories, e.g. *encrypt passwords* is specific for *cryptography*, *passwords* and *protect security secrets*. A category without more generic categories is a *base category*. All recursively resolved specific categories for category *c* are the *subcategories* of *c*. The category itself belongs to its subcategories. The taxonomy allows addressing the problem that related requirements in different sources can go to different subcategories. None of the subcategories may be popular enough to show in the results. However, a generic category aggregates the popularity of its subcategories and may then appear in the results.

Many sources refer to other sources, e.g. Cybersecurity Label references selected requirements from ETSI EN 303 645 (NCSC-FI, 2021; ETSI, 2020). Reading the sources, it is clear that the authors still

use their knowledge to choose the referenced requirements, thus contributing to the accumulated knowledge.

2.2 Randomness of Categories

Let C be the population of security requirement categories, S the set of sources, and C_s the set of categories used in source s . The first research question translates to whether there is a consensus on the requirement categories in the sources. The null hypothesis is that source categories C_s appear to be randomly chosen from C . Surely no author writes random requirements, but without common preferences, the sources do not correlate and categories appear to be randomly chosen. To test the null hypothesis, we calculate the probability of sharing categories by chance. The probability of randomly choosing any category for source s is $p_s = \frac{|C_s|}{|C|}$. The population C is larger than the observed categories if there exists requirement categories authors considered but ultimately none included. Let p_n be the probability of category being shared exactly by n sources. Its expected value is calculated by summing up the probabilities of all source combinations where a category is present n times,

$$E(p_n) = \sum_{K \in S^n} \prod_{s \in S} \begin{cases} p_s, & s \in K \\ 1 - p_s, & s \notin K \end{cases}, \quad (1)$$

where K iterates through values of $S^n = \{A \in P(S) \mid |A| = n\}$, the set of combinations of exactly n sources. $P(S)$ is the power set of S . The factor for each source is either p_s or $1 - p_s$ depending whether the category is present in the source or not. Assuming the category probabilities are independent, the expected value for the number of categories shared by n sources c_n is

$$E(c_n) = |C|E(p_n). \quad (2)$$

Once we have the real data from the sources, we can calculate the actual category distribution and see if it matches the random distribution or not.

3 RESULTS

We identified 16 requirement sources that matched our criteria. The sources cover many IoT domains: consumer, industrial control systems (ICS), vehicle, medical, and generic IoT. The sources contain 1525 original security requirements, 2243 individual requirements, and 269 categories. Table 1 lists the sources and their industry domain, requirement

Table 1: Requirement sources. For each source, the table gives domain, requirement count, category count, and proportion of categories shared by other sources. Row ‘‘Source average’’ gives the source average values. Row ‘‘Total’’ gives the total requirement count, total category count, and proportion of all categories shared by a different number of sources.

Source	Domain	Requirements	Security categories	not shared	shared by 2-5	shared by 6-9	shared 10-
C2 IoT Security Baseline (CSDE, 2019)	Generic	30	28	0	21%	54%	25%
CTIA Cybersecurity Test Plan (CTIA, 2021)	Generic	42	30	7%	40%	20%	33%
Cybersecurity Label (NCSC-FI, 2021)	Consumer	26	23	0	17%	48%	35%
ENISA Security Baseline (ENISA, 2017)	Generic	129	91	4%	40%	40%	16%
ETSI EN 303 645 (ETSI, 2020)	Consumer	90	58	5%	33%	40%	22%
FDA Draft Guidance (FDA, 2018)	Health	132	77	1%	40%	42%	17%
GSMA IoT Security Assessment (GSMA, 2018)	Generic	437	117	7%	47%	32%	14%
ISASecure CSA (ISASecure, 2019)	ICS	351	109	5%	46%	36%	14%
IMDA IoT Cyber Security Guide (IMDA, 2020)	Generic	95	80	6%	34%	41%	19%
Internet Society IoT Trust (IS, 2017)	Generic	85	60	2%	42%	40%	17%
IoT Security Initiative (IoTSI, 2018)	Generic	67	49	4%	43%	37%	16%
IoTSC Security Compliance (IoTSCF, 2020)	Generic	366	125	6%	48%	34%	12%
IoT Acceleration Consortium (IoTAC, 2016)	Generic	63	30	10%	37%	33%	20%
NCSC Health Requirements (NCSC-FI, 2019)	Health	230	104	17%	40%	32%	11%
NHTSA Best Practises (NHTSA, 2020)	Vehicle	73	42	5%	40%	38%	17%
NISTIR 8259a (NIST, 2020)	Generic	27	18	0	44%	11%	44%
Source average		140	65	5%	38%	36%	21%
Total		2243	269	23%	51%	20%	6%

counts, and category counts. The requirement counts per source vary from 26 to 437 and category counts from 18 to 125 with averages 140 requirements and 65 categories. A source with many requirements also has many categories. The table shows separately the count of categories unique for a source and shared between 2-5, 6-9, and 10 or more sources. On average, a source has 5% unique categories. From all categories, 74% are in five or fewer sources.

3.1 Hypothesis Verification

The null hypothesis is that the source categories are randomly chosen. Testing of the hypothesis requires a value for the category population size $|C|$ which minimum value is 269, the number of categories we observed in the sources. Different sizes were checked by using equation 2 to calculate the expected $E(c_n)$ for $n = 1, 2, \dots, 15$. The values were compared with the observed number of shared categories in the sources. The smallest cumulative difference was with $|C| = 291$. Table 2 shows the observed number of categories, expected value, standard deviation, and p-value for the number of categories shared by n sources with the population size 291. For this and other population counts the observed and expected values meet on a few values of n , but are significantly different on most. It is highly unlikely that the observed category distribution results randomly. We reject the null hypothesis, there exists a set of common requirement categories shared by the sources.

Table 2: Observed number of categories c_n , expected values $E(c_n)$, standard distribution $SD(c_n)$, and p-values for categories shared by n sources and population size $|C_n| = 291$. Symbol ϵ stands for p-values smaller than 0.001.

n	Observed c_n	$E(c_n)$	$SD(c_n)$	p
0		4	2.0	
1	62	21	4.6	ϵ
2	50	50	7.1	0.99
3	38	70	8.4	ϵ
4	35	67	8.2	ϵ
5	14	45	6.7	ϵ
6	20	22	4.7	0.61
7	20	8	2.9	ϵ
8	8	2	1.5	ϵ
9	6	1	0.7	ϵ
10	6	0	0.3	ϵ
11	6	0	0.1	ϵ
12	2	0	0.0	ϵ
13	1	0	0.0	ϵ
14	1	0	0.0	ϵ
15	0	0	0.0	1.00
16	0	0	0.0	1.00

The common security requirement categories from the sources are presented next. For readability, they are divided into product security requirements and life-cycle requirements.

3.2 Product Security Requirements

Product requirements target the IoT device and the supporting system. Table 3 shows the product security base categories and subcategories covered by

Table 3: Common product requirement categories. The table shows base categories, subcategories covered by at least 10 sources, the number of covering sources, and the proportion of subcategories from all categories. Subcategories are prefixed by a dash. As a subcategory can be a specific category in many categories, the sum of proportions exceeds 100%.

Category	Sources	Prop.	Category	Sources	Prop.
Security design	16	9.3%	Security hardware	13	1.9%
– security architecture	16	8.9%	– tamper protection	11	0.4%
– subsystems	10	2.2%	Backend security	9	1.9%
– least privilege	15	1.5%	Cryptography	13	5.9%
Secure programming	7	1.9%	Data protection	16	14%
Delivery & deployment	14	4.8%	– protect secrets	11	0.7%
– deployment integrity	14	4.5%	– protection in transit	15	1.1%
– no global cred.	10	0.4%	– comm. standards	13	0.4%
Administration	13	5.6%	– decommission	11	0.7%
– secure config.	10	3.7%	Service availability	12	2.2%
Interface security	16	9.3%	Failure security	6	1.1%
– min. attack surface	13	4.8%	Audit logging	11	1.9%
– no unused interfaces	11	0.4%	Intrusion detection	15	5.2%
– validate input	10	1.1%	– software integrity	11	0.7%
Authentication	16	13%	Incident response	8	1.9%
– passwords	12	4.1%	System updates	16	5.9%
– auth. brute force	10	0.4%	– update security	14	0.4%
– component auth.	14	2.2%	Usability of security	10	2.2%
Access control	10	2.6%			

at least 10 sources, the number of covering sources, and the proportion of subcategories from all categories. The categories covered by all sources are *Security design*, *Interface security*, *Authentication*, *Data protection*, and *System updates*. The subcategories of both *Authentication* and *Data protection* represent over 10% share of all categories.

The next paragraphs explain the product requirement categories as described in the requirement sources.

Security design considers the whole IoT system holistically and includes the **security architecture**. The system is made up of **subsystems** and their connections. The design follows the security design principles, such as the **least privilege**. **Secure programming** practices are enforced as relevant for the used languages, frameworks, compilers, etc. **Delivery & deployment** of the system must be secure. The **deployment integrity** requires the installation is not tampered with, the environment meets security requirements, and the system is hardened and secure after the installation. There must be **no global credentials**, such as shared passwords. **Administration** functions are protected, especially the remote administration interfaces. The **secure configuration** of accounts, roles, networks, and other critical settings is ensured.

Interface security deflects attacks at the system boundary, which often is the network interface. De-

fence is easier by **minimizing the attack surface**, such as network ports and open services, and having **no unused interfaces**. One should **validate input** from untrusted sources to avoid disruption by malicious input. **Authentication** by **passwords** or other means establishes the identity of the users. Password guessing and other **authentication brute force** attacks are mitigated. Proper **component authentication** identifies the components of the system. **Access control** limits the actions available for users and components. **Security hardware** provides security features, such as cryptographic functions, memory isolation, and trusted platform modules (TPMs). Equipment located outside secure premises requires **tamper protection**. **Backend security** is an essential part of IoT system security. Backend is often in a cloud and there is a related web service, which must also be secured. **Cryptography** should be based on established standard algorithms. The used cryptographic keys must be securely generated, stored, and updated. **Data protection** must include all critical data, especially private information. The system must **protect secrets** such as keys and provide data **protection in transit** over networks and other untrusted media, preferably using secure **communication standards**. Finally, the **decommission** of an IoT device includes the destruction of critical data. **Service availability** requirements are strict for critical infrastructure, e.g. ICS systems, but all IoT devices must tolerate power

and network outages. **Failure security** means that if the system fails due to internal or external reasons, the system does not enter an insecure state or disclose critical data.

Audit logging stores the important events for later analysis. **Intrusion detection** system (IDS) monitors the system for malicious activity. Unauthorized software or firmware changes are detected to maintain the **software integrity**. An intrusion prevention system (IPS) actively mitigates attacks, such as denial of service (DOS) attacks. **Incident response** should be supported by the system, e.g. by facilitating backup restore or providing means to check system integrity. **System updates** are required for security fixes and enhancements after deployment while maintaining **update security** to ensure that the applied updates are not compromised. **Usability of security** ensures administrators and users know how to use the system securely, including instructions for deployment and use. Security-critical decisions should be avoided. Idle sessions should be locked or terminated after a timeout.

It is notable and surprising that the category for **privacy** was only covered by nine sources, thus it did not make it into the common categories.

3.3 Life-cycle Security Requirements

Table 4 shows the common categories for IoT product life-cycle security requirements. The most common categories are *Security requirements*, *Security standards* and *Vulnerability management* covered by 14 sources. The next paragraphs explain the life-cycle requirement categories as described in the sources.

Vendor security is required to produce and maintain secure products. The management and other personnel must know the best practices and be committed to follow them. **Policies & laws**, e.g. general data protection requirements (GDPR), must be honoured to avoid legal and contract issues. **Development process** activities and tools support the production of secure products. The **external components** acquired through the supply chain must be carefully managed. Testing must include **security tests**. **Security requirements** are derived from customer needs and **risk analysis**. **Security standards** relevant for the product must be followed, e.g. the **communication standards**. **Vulnerability management** is about collecting information on the relevant security vulnerabilities and responding to it properly and promptly on time, especially in case of publicly **known vulnerabilities**. **User communication** keeps users and administrators informed about security threats, mitigations, updates, and other important events.

3.4 Coverage of the Security Categories

There are 25 base categories combining the product and life-cycle security requirements. Table 5 gives the coverage of the categories in the sources. A source covers between 13 to 25 of the base categories, the average is 19. The average number of subcategories per source varies between 9.1 for *Data protection* and 1.3 for *Failure security*

To get a complementary view to the security requirements, the requirements were assigned into the NIST Cybersecurity Framework functions *Identify*, *Protect*, *Detect*, *Respond*, and *Recover* (NIST, 2018). Requirements are assigned into the function they support, e.g. category *Data protection* requirements for function *Protect*. The result is shown in Table 6. The requirements supporting *Identify* function are for risk analysis, secure design, and management of the supply chain. *Protect* is supported by requirements for authentication, access-control, interface protection, data protection and management, etc. This function receives the majority of categories. *Detect* is supported by requirements for logging, tamper protection, and intrusion detection. *Respond* is supported by requirements for vulnerability management and incident response. *Recovery* is supported by requirements for backup restore and securing of the systems after incidents. About 50 requirement categories for vendor security, development process, and security standards were not assignable to any framework functions.

3.5 Comparison to Other Studies

We compared the requirements categories with some other studies. Momenzadeh et al. collected 56 best practices for security analysis of two consumer IoT devices and covered 16 base categories (Momenzadeh et al., 2020). The practices were divided into categories privacy and authentication, system operation, device policies, vulnerability mitigation, and device operation. Any best practices not observable from the products were dropped, thus excluding vendor security, backend security and secure programming. Stellios et al. presented a set of ICS security controls based on common attack patterns and risk analysis and covered 13 base categories (Stellios et al., 2018). The controls aim to reduce threat level, reduce vulnerability level, or reduce potential impact of connectivity. The focus is also on the product security requirements, although security testing is mentioned from life-cycle requirements. Many of the requirements are for administration. There are no requirements for backend, cloud, web, secure programming,

Table 4: Common life-cycle requirement categories. The table shows base categories, subcategories covered by at least 10 sources, the number of covering sources, and the proportion of subcategories from all categories. Subcategories are prefixed by a dash. A subcategory can be a specific category in many categories.

Category	Sources	Prop.	Category	Sources	Prop.
Vendor security	8	3.7%	Security standards	14	2.2%
Policies & laws	10	6.3%	– comm. standards	13	0.4%
Development process	12	8.6%	Vuln. management	14	3.7%
– ext. components	10	1.1%	– known vuln.	10	0.7%
– security tests	10	4.1%	User communication	12	1.9%
Security requirements	14	4.5%			
– risk analysis	11	0.7%			

Table 5: Coverage of the security base categories in the sources. For each base category, the table shows the total subcategory count, count per source, and average count per source. The last row shows the number of covered base categories per source.

	Cats.	(CSDE, 2019)	(CTIA, 2021)	(NCSC-FI, 2021)	(ENISA, 2017)	(ETSI, 2020)	(FDA, 2018)	(GSMA, 2018)	(ISASecure, 2019)	(IMDA, 2020)	(IS, 2017)	(IoTSI, 2018)	(IoTTF, 2020)	(IoTAC, 2016)	(NCSC-FI, 2019)	(NHTSA, 2020)	(NIST, 2020)	Avg.
Vendor security	10				3		2	2		2		3	3	5	5		3.1	
Policies & laws	17		1	2	8	6		12		1	9	1	9		5		5.4	
Development process	23	3			6	1	10	7	17	4	4	7	9		10	10	7.3	
Security requirements	12	1		1	7	1	6	5	5	2	3	1	3	5	4	4	3.4	
Security design	25	2	3	1	7	3	6	9	11	6	1	9	9	4	10	2	5.3	
Security standards	6	2	2	1	3	1	2	3	2	2	2		5		3	1	2.2	
Secure programming	5				1	1		2	3		1	1	3				1.7	
Delivery & deployment	13		1	1	2	2	7	8	4	3	2	1	5	1	5	1	3.1	
Administration	15	3	2	2		4	2	4	7	4	2	3	3		7	3	3.5	
Interface security	25	2	1	4	9	6	5	10	12	9	2	7	15	1	11	5	6.2	
Authentication	35	2	8	2	8	6	11	14	12	12	8	7	16	1	14	1	7.8	
Access control	7	1	2		1		3	2	4	5			2	3		1	2.4	
Security hardware	5	2	1		3	2	1	3	2	1	1	2	3	1	1		1.8	
Backend security	5				3		2	4		1	3	2	4		3	1	2.6	
Cryptography	16	3	1		7	4	2	8	5	8	1	3	13		6	1	4.8	
Data protection	38	5	5	3	17	10	5	22	7	9	16	5	16	3	18	2	9.1	
Service availability	6	1			1	3	2	3	3	1	1	1	3	1	1		1.8	
Failure security	3		1		2			1	2			1		1			1.3	
Audit logging	5	2	2		1		2	1	4	2				3	2	1	1.9	
Intrusion detection	14	1	3		6	4	7	6	7	8	1	1	9	3	5	3	4.4	
Incident response	5				1		3	1	2	2			1		3		1.8	
Vuln. management	10	2		4	4	4	4	4	7	4	5	1	5	3	4	6	4.1	
System updates	16	2	4	4	6	9	4	4	4	5	7	1	10	1	7	1	4.6	
User communication	5			1	1	2		1	3	3	4		3	1	3	1	2.0	
Usability of security	6			2		3	2	1	2	2	1		3	2	2		2.0	
Coverage		16	15	13	23	19	20	25	23	22	21	18	23	16	23	15	13	19

or incident response. Tange et al. and Hansch et al. researched the security requirements for industrial IoT and both covered 16 base categories (Tange et al., 2020; Hansch et al., 2019).

Overall the requirement counts in the studies are comparable to the sources with a small number of requirements. The number of covered categories is on the low side. The studies are often vague on the details of the requirements, which makes categorization

challenging. It is hard to compare them with the other sources.

4 DISCUSSION

We collected security requirements from 16 different sources and divided them into 25 base categories and 244 subcategories. Based on the frequency of the cat-

Table 6: Assignment of requirements into NIST Cybersecurity Framework functions. For each function, the table shows the requirement category count, count per source, and average count per source.

	Cats.	(CSDE, 2019)	(CTIA, 2021)	(NCSC-FI, 2021)	(ENISA, 2017)	(ETSI, 2020)	(FDA, 2018)	(GSMA, 2018)	(ISASecure, 2019)	(IMDA, 2020)	(IS, 2017)	(IoTISI, 2018)	(IoTISF, 2020)	(IoTAC, 2016)	(NCSC-FI, 2019)	(NHTSA, 2020)	(NIST, 2020)	Avg.
IDENTIFY	61	5	5	4	23	11	15	29	20	11	14	11	24	9	22	8	3	13.4
PROTECT	163	19	27	17	54	43	43	75	63	53	38	29	74	17	68	17	15	40.8
DETECT	21	4	5		8	5	11	8	12	11	3	2	10	6	7	5	2	6.6
RESPOND	22	4	2	4	7	5	10	8	14	9	6	1	7	7	10	8	2	6.5
RECOVER	3						3	1	1	2					2			1.8

egories shared by the sources, the sources do agree on common security requirements. The categories covered in all sources are *Security design*, *Interface security*, *Authentication*, *Data protection*, and *System updates*. *Intrusion detection* was covered by all but one source. The categories describe a well-designed product with a secured interface, authentication of users, protection of critical data, and mechanism for security updates. Detection of intrusions only makes sense if someone can react and perform incident response, unlikely for consumer IoT. Unexpectedly, the category *privacy* was not frequent enough to be a common category. Perhaps the protection of private data is included in general data protection or considered irrelevant for a domain like ICS. The most common categories are product-related. The life-cycle requirements may have been considered proprietary to vendors or out of scope for IoT security requirements. The top life-cycle categories are *Security requirements*, *Security standards*, and *Vulnerability management*. The first two support the design and implementation of a secure product and the latter is essential for maintaining security through product lifetime. The agreement on security requirements may partly be caused by the source authors referencing other sources. However, the references appear to be made selectively, thus contributing to the shared knowledge.

The selection of categories and subcategories is a judgement call. We tried to be faithful to how the requirements were presented in the sources. Many of the categories have a lot of subcategories, which may reflect both complexity or disagreement among the sources. *Data protection* has the most subcategories, 38, and an average of nine subcategories per source. Many categories are present only in one or a few sources resulting in a long tail of odd security requirements. The more requirements a source has, the more unique categories it tends to introduce.

A total of 10 sources are for generic IoT and six for specific domains. The common categories are

generic and applicable in all IoT domains. Categories in the domain-specific sources do not appear to differ much from the generic sources. Based on this, the need for domain-specific IoT requirements looks limited.

We also mapped the security requirements into NIST Cybersecurity Framework. The majority of the applicable requirements support the framework function *Protect*. There was some support for functions *Identify*, *Detect*, and *Respond*, but only marginal for *Recovery*. The security requirements are focused on incident prevention rather than response.

5 CONCLUSIONS

In this study, the security requirements in 16 standards, guidelines and best practices were divided into requirement categories. A set of common categories was identified, with all sources covering *Security design*, *Interface security*, *Authentication*, *Data protection*, and *Updates*. The common categories are made up of many subcategories and many different security requirements. The majority of requirement categories are only present in five or fewer sources and many are unique to a single source. Thus, despite the shared high-level categories, the selection of a requirement source has a big impact to the individual requirements an IoT project must fulfil. Adding new requirement sources likely brings in many new requirements.

Consolidation of IoT security requirements seems desirable and possible, as even while the sources disagree on details, they are aligned in the high-level security targets.

ACKNOWLEDGEMENTS

This work is done in the SECREDAS project funded by Horizon 2020 programme (grant agreement nr. 783119) and by Business Finland.

REFERENCES

- CSDE (2019). The C2 Consensus on IoT Device Security Baseline Capabilities. , Council to Secure the Digital Economy.
- CTIA (2021). Cybersecurity Certification Program for IoT Devices, Version 1.4. , CTIA Certification.
- ENISA (2017). Baseline Security Recommendations for IoT in the context of Critical Information Infrastructures. , European Union Agency For Network And Information Security.
- ETSI (2020). Cyber Security for Consumer Internet of Things: Baseline Requirements v2.1.1. ETSI EN 303 645, ETSI.
- FDA (2018). Content of Premarket Submissions for Management of Cybersecurity in Medical Devices, Draft Guidance for Industry and Food and Drug Administration Staff. , US Food and Drug Administration.
- GSMA (2018). GSMA IoT Security Assessment CLP.17, Version 3.0. , GSM Association.
- Hansch, G., Schneider, P., Fischer, K., and Böttinger, K. (2019). A unified architecture for industrial iot security requirements in open platform communications. In *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 325–332.
- IMDA (2020). (IoT) Cyber Security Guide, Version 1. , Singapore Info-communications Media Development Authority.
- IoTAC (2016). IoT Security Guidelines, ver. 1.0. , IoT Acceleration Consortium, Japan.
- IoTSEF (2020). IoT Security Compliance Questionnaire. , IoT Security Foundation.
- IoTSEI (2018). IoT Security Design Best Practises. , IoT Security Initiative.
- IS (2017). IoT Security & Privacy Trust Framework v2.5. , The Internet Society.
- ISASecure (2019). Component Security Assurance - ISASecure certification scheme, Version 4.3. CSA-100, ISA Security Compliance Institute.
- Momenzadeh, B., Dougherty, H., Rimmel, M., Myers, S., and Camp, L. J. (2020). Best practices would make things better in the IoT. *IEEE Security Privacy*, 18(4):38–47.
- NCSC-FI (2019). Information security and data protection requirements for social welfare and healthcare procurements, v.1.0.6. , National Cyber Security Centre Finland.
- NCSC-FI (2021). The Finnish Cybersecurity Label. , National Cyber Security Centre Finland.
- NHTSA (2020). Cybersecurity Best Practices for the Safety of Modern Vehicles, Draft 2020 Update. , US National Highway Traffic Safety Administration.
- NIST (2018). Framework for Improving Critical Infrastructure Cybersecurity Version 1.1. , US National Institute of Standards and Technology.
- NIST (2020). IoT Device Cybersecurity Capability Core Baseline. NISTIR 8259A, US National Institute of Standards and Technology.
- Sayegh, E. (2021). Peloton breach reveals a coming iot data winter. *Forbes*.
- Stellios, I., Kotzanikolaou, P., Psarakis, M., Alcaraz, C., and Lopez, J. (2018). A survey of iot-enabled cyberattacks: Assessing attack paths to critical infrastructures and services. *IEEE Communications Surveys Tutorials*, 20(4):3453–3495.
- Tange, K., De Donno, M., Fafoutis, X., and Dragoni, N. (2020). A systematic survey of industrial internet of things security: Requirements and fog computing opportunities. *IEEE Communications Surveys Tutorials*, 22(4):2489–2520.