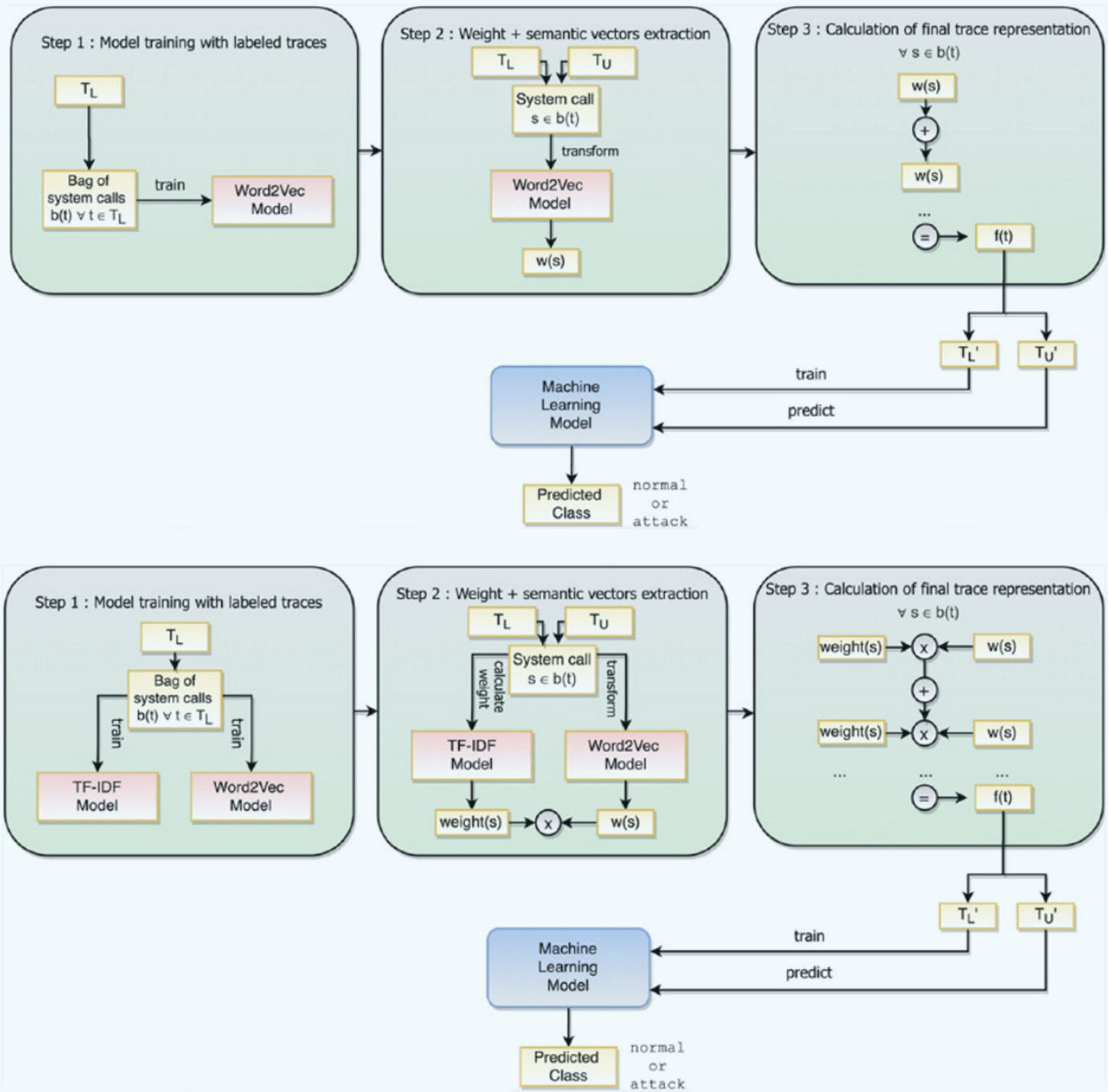
ISSN 2694-1015 (Online)
Volume 1 I Issue 2 I December 2020

Journal of Surveillance, Security and Safety

JSSS



OAE Publishing Inc.

EDITORIAL BOARD

GENERAL INFORMATION

About the Journal

Journal of Surveillance, Security and Safety (JSSS) is an open access, peer-reviewed, quarterly online journal which provides a forum for the publication of papers addressing the variety of theoretical, methodological, epistemological, empirical and practical issues concerns reflected in the field of information security, cyber security, machine learning, emerging technologies, and their applications. In particular, the journal encourages articles in the following areas:

AI-based surveillance and security
Privacy protection based on machine learning
Security of machine learning algorithms
Deep learning for attack and defense
Database security
Data-driven cybersecurity incident prediction
Big data security
Cloud/fog computing security
Outsourcing and crowdsourcing security
Security and privacy in pervasive/ubiquitous computing
Cyber-physical systems security
Security, privacy and resilience in critical infrastructures
Multimedia security
Wireless network security
Social networks and IoT security
Information hiding, forensics and security
Theory and applications of cryptography
Identity management, authentication and access control
Security policies, models and architectures
Electronic commerce security
Blockchain and finance security
Intrusion detection
Phishing and spam prevention
Biometrics
Regulation of the security industry
Risk analysis, security measures and management
Evaluations of security measures

Information for Authors

Manuscripts must be prepared in accordance with Instructions to Authors. Please check: https://jsssjournal.com/pages/view/author_instructions for details. All manuscripts must be submitted online at: www.jsssjournal.com/login.

Copyright

Permissions

Disclaimer

Published by

OAE Publishing Inc.
245 E Main Street ste115, Alhambra, CA 91801, USA
Website: www.oaepublish.com

Contacts

E-mail: editorial@jsssjournal.com
Website: www.jsssjournal.com

Journal of Surveillance, Security and Safety

CONTENTS

Calderoni *et al. J Surveill Secur Saf* 2020;1:106-18
DOI: 10.20517/jsss.2019.01

Journal of Surveillance, Security and Safety

Original Article

Open Access

Forge-resistant radio-frequency identification tags for secure internet of things applications

Luca Calderoni[1], Dario Maio[1], Luciano Margara[1], Luca Spadazzi[2]

[1]Department of Computer Science and Engineering, University of Bologna, Cesena 47522, Italy.
[2]Lab51 srl, Cesena 47522, Italy.

Correspondence to: Prof. Luca Calderoni, Department of Computer Science and Engineering, University of Bologna, via dell'Università, 50, Cesena 47522, Italy. E-mail: luca.calderoni@unibo.it

How to cite this article: Calderoni L, Maio D, Margara L, Spadazzi L. Forge-resistant radio-frequency identification tags for secure internet of things applications. *J Surveill Secur Saf* 2020;1:106-18. http://dx.doi.org/10.20517/jsss.2019.01

Received: 13 Dec 2019 **First Decision**: 1 Feb 2020 **Revised**: 10 Feb 2020 **Accepted**: 31 Mar 2020 **Available online**: 29 Oct 2020

Academic Editor: Michael G. Pecht **Copy Editor**: Jing-Wen Zhang **Production Editor**: Jing Yu

Abstract

Aim: Internet of Things (IoT) represents a key aspect within several application domains, and it enables growing opportunities for both organizations and end-users. Radio-frequency identification tags are probably the most relevant enabling solution for ubiquitous IoT systems and are often seen as a prerequisite for IoT itself. In this study, we analyzed one of the most promising radio-frequency identification tags to determine whether or not it represents a viable solution for secure IoT applications.

Methods: The study was conducted relying on an Android OS application developed within our laboratories, which helped us to inspect the chip and describe its logical data structure. We studied the capabilities of the tag in relation to the application protocol data unit it supports, and we described the cryptographic protocols with which it is equipped.

Results: This tag is resistant to forging activities, and it also preserves confidentiality and authenticity on exchanged data. We discussed several known privacy and security patterns that may be addressed relying on the tag we focused on and we underlined some deficiencies concerning chip cloning attack. Again, secure dynamic messaging and mirroring allow the surpassing of several privacy limitations.

Conclusion: In this paper we investigated the capabilities of the *NT4H2421Gx* tag. The deep Android inspection performed on the tag showed that it represents an option to rely on when we need to design secure IoT applications.

Keywords: Radio-frequency identification, NFC, internet of things, cryptographic protocols

1 INTRODUCTION

Internet of Things (IoT) has exploded in recent years, and the related security aspects are increasingly relevant[1,2]. Radio-frequency identification (RFID) represents the most adopted solution within the IoT domain[3,4]. The logistics industry is one of the earliest adopters of IoT and RFID solutions[5], while these technologies are now used in several application contexts, such as military and defense applications, supply chains, food industry and so forth. As an example, RFID tags may be applied to manage inventories, to reduce overstocks and to avoid understocks as well as to track the overall lifecycle of a product[6].

More recently, RFID tags have also been used for different applications, such as localization and personal identification. For example, electronic machine readable travel documents are equipped with RFID tags[7]. As this feature enables several cryptographic protocols to be applied during the communication between the tag and the reader, it also makes it possible to deliver automated border controls in crucial areas such as international airports. At the same time, localization and identification procedures based on RFID also imply privacy and traceability issues for the tag bearer[8-10].

Thus, the combination of RFID and cryptography is widely studied[11-14], and paving the way for a number of pervasive and secure applications. Among them, those aimed at preventing forgery and counterfeiting of trademark products represent a significant slice of the application sector. In recent years, the scientific community has therefore dedicated significant efforts to the design of techniques aimed to prevent malicious attacks against RFID technology[15-18]. Consequently, several efficient cryptographic protocols were proposed to deliver high-quality protection mechanisms for RFID-based applications.

The RFID industry tries to adapt its products so they can fit this rapid evolution and continues to produce new tags with smarter capabilities. Each RFID tag has different features, including the supported cryptographic protocols, the amount of data that it is able to store, the set of commands it can deal with and so forth. The design of a secure IoT application relying on RFID technology should be thus preceded by an in-depth study of tag capabilities. In this study, we focused on *NT4H2421Gx*[19], a recent RFID tag released by *NXP Semiconductors*, and we investigated its features extensively. The results showed that *NT4H2421Gx* represents a valid and promising solution for a wide number of secure IoT applications.

2 METHODS

In this section, we described the features of *NT4H2421Gx*. After a brief introduction to the general specifications of the tag, we investigated in depth its logical data structure, its application protocol data unit (APDU) and its core functionalities. Finally, we proposed a high-level comparison between this tag and other related ones.

The NXP's *NT4H2421Gx* tag is fully compliant with the *NFC Forum Type 4 IC* specification and relies on the *ISO/IEC 14443-4* contactless proximity protocol. The file system is compliant with *ISO/IEC 7816-4*[20]. The APDU is based on *ISO/IEC 7816-4* as well, while it preserves only three of the native commands. Each command included in the command set is tag specific.

2.1 Hardware layer

Contactless smart cards with microprocessors incorporate their own operating system, which is usually burned into the ROM module at the production stage. The tasks of the operating system are data transfer from and to the smart card, command sequence control, APDU interpretation, file management and

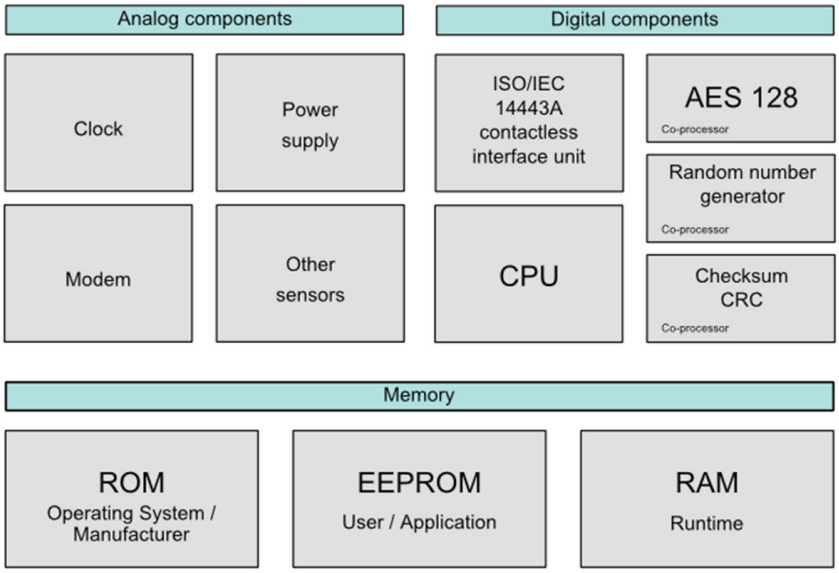


Figure 1. A high-level hardware block diagram of the *NT4H2421Gx* tag

cryptographic algorithm execution (e.g., encryption, authentication)[21]. Concerning *NT4H2421Gx*, a high-level block diagram depicting its hardware components is provided in Figure 1.

Usually, a command processing sequence within a smart card operating system undergoes the following flow. At the *physical layer*, commands sent from the reader to the tag are received through the radio frequency interface, according to *ISO/IEC 14443-2A*. The packets are processed at the *transport layer* according to *ISO/IEC 14443-3A*: error detection and correction are performed by the I/O manager, which relies on the CRC co-processor. If the packet is deemed correct, its payload is extracted and processed at the *application layer*, relying on *ISO/IEC 7816-4* or proprietary APDU commands. When secure messaging applies, the payload is decrypted or checked for integrity. These procedures are enhanced by the AES and RNG co-processors. When the APDU manager is not able to recognize the command, the return code manager generates the appropriate return code and sends it back to the reader. Conversely, if a valid command is received, the system executes the instructions which correspond to the command code, according to the APDU. When the command implies some access to the EEPROM, this is performed exclusively by the file management system and the memory manager, which convert all symbolic addresses into the corresponding physical addresses of the memory area. The file manager is also responsible for verification of access conditions, depending on the addressed data.

2.2 Logical data structure

Concerning the file system, *NT4H2421Gx* complies with *ISO/IEC 7816-4*. Specifically, it is equipped with a master file (MF), a dedicated file (DF) and three elementary files (EF). The logical data structure mounted on the tag we focused on is depicted in Figure 2.

The first file is also known as the *capability container* (CC) file and it is formatted in accordance with *NFC Forum* specifications[22]. This file specifies the mapping version and the maximum size of command APDU and response APDU data size. Moreover, this file contains some metadata concerning the other two files included in the user memory. For each of them, this file specifies the name of the file, the overall byte size and the access conditions which need to be met to access the file. The "Results" section provides a deep look at the CC file.

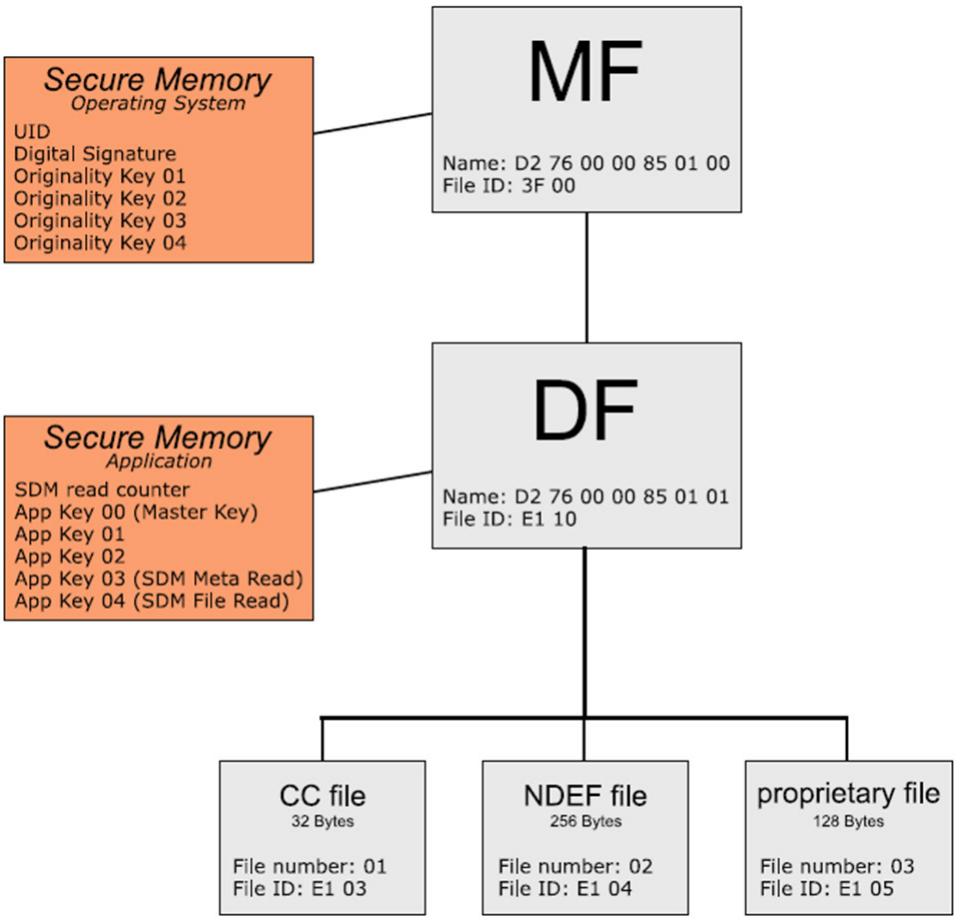


Figure 2. The file system mounted in the user memory. The three *elementary files* listed under the *dedicated file* are *standard data files*, according to *ISO/IEC 7816-4*. MF: master file; DF: dedicated file; CC: capability container; SDM: secure dynamic messaging; UID: unique tag identifier; NDEF: NFC data exchange format; NFC: near field communication

The second file is also known as the *NDEF* file and contains an NDEF-formatted message[23]. NDEF is a lightweight, binary message format that can be used to encapsulate one or more application-defined payloads of arbitrary type and size into a single message construct. Each payload is composed of a type, length, and optional *id*. Just as an example, identifiers may be represented by URIs, MIME media types, or other NFC-specific types. This file is also designed to support *secure dynamic messaging* (SDM) and *data mirroring*. These options extend the security and privacy features offered by this tag and will be discussed in the next sections.

The third file is a proprietary NXP file which is read- and write-protected and contains raw data. At the production stage, access to this file is restricted using two different application keys, one for reading operations and one for writing operations. This condition is better exemplified in the "Results" section.

The RFID device also includes nine cryptographic keys, designed to be used as advanced encryption standard (AES) keys[24]. Four keys are provided at the tag level (MF). They are also referred to as *originality keys*. The other five keys are instead included at the application level (DF) and are referred to as *application keys*. Originality keys are stored in ROM and may never be removed or updated after chip production. Conversely, application keys are part of the user memory (EEPROM) and may be updated to customize the tag for application-specific scenarios. Each of these nine keys may be used to perform an authentication procedure between the tag and a reader. Moreover, to update any of the app keys, a successful

Table 1. AES keys installed on NT4H2421Gx tag

Key	Length	Location	Key n	Update	Authentication	Notes
Originality key 1	128 bits	ROM	0x01	×	√	
Originality key 2	128 bits	ROM	0x02	×	√	
Originality key 3	128 bits	ROM	0x03	×	√	
Originality key 4	128 bits	ROM	0x04	×	√	
Application key 1	128 bits	EEPROM	0x00	√	√	App master key
Application key 2	128 bits	EEPROM	0x01	√	√	
Application key 3	128 bits	EEPROM	0x02	√	√	
Application key 4	128 bits	EEPROM	0x03	√	√	SDM meta read
Application key 5	128 bits	EEPROM	0x04	√	√	SDM file read

While App Master Key is always identified by code 0x00 at the dedicated file level, SDM-related keys may be identified by each of the application keys (i.e., it is not mandatory to use key 0x03 and 0x04 as reported in this table). SDM: secure dynamic messaging

authentication through the first application key is required. This key is also referred to as *App Master Key*. A complete list of the aforementioned keys is provided in Table 1.

Finally, it is important to point out that the tag ROM also contains the *unique tag identifier* (UID), composed of 7 bytes, and a 56-byte *digital signature,* which was computed by NXP at the production stage and burned in the memory. This digital signature lays at the basis of the strong anti-forging functionalities provided by the *NT4H2421Gx* tag and will be discussed in the next section.

2.3 Application protocol data unit

An APDU consists of the instruction set used by the reader and the tag during communication. Each procedure that is performed during communication relies on a combination of APDU commands. APDU instructions are divided into *command APDUs* and *response APDUs*. The former ones are sent by the reader to the tag while the latter are sent back by the tag to the reader.

NT4H2421Gx APDU is based on the *ISO/IEC 7816-4* standard. However, the majority of available commands are proprietary and are programmed through original *ISO/IEC 7816-4* command wrapping. Specifically, only three of the native commands are preserved.

The complete *NT4H2421Gx* command set is provided in Table 2. Please note that some of the listed commands are composed of more than one part. For instance, the *GetVersion* command is divided into *GetVersion part1*, *GetVersion part2* and *GetVersion part3*. These details do not add much to the discussion on the subject and are therefore omitted for brevity.

2.4 Comparison

NT4H2421Gx is a robust and versatile tag and provides a wide range of desirable features within the IoT domain. As summed up in Table 3, this tag was introduced by NXP to surpass several limitations that afflicted tags belonging to older generations. NTAG is the market-leading portfolio of NFC tag solutions for the consumer and industrial segments of IoT. These tags offer different levels of security and different functionalities as well, to address a wide range of applications.

NT4H2421Gx supports NDEF-formatted messages to be stored in the user memory. NDEF records may be combined with UID mirroring, UID randomization and SDM to cover a broad range of user requirements, including privacy preservation. Thanks to several co-processors, this tag also provides authentication functionalities and secure messaging. Both of them rely on AES-128 cryptography. Memory access is subject to a mixture of user-driven and manufacturer-driven permissions and relies on AES-128 authentication as well. Forging attempts are averted by the manufacturer's digital signature (56 bytes), which is computed against the UID at the production stage and is embedded into the tag.

Table 2. NT4H2421Gx command set

Category	Command	Class	Description
Basic r/w functionalities	ISOSelectFile	ISO/IEC 7816-4	Select MF, DF or EF
	ISOReadBinary	ISO/IEC 7816-4	Read data from a data file (EF)
	ISOReadBinary	ISO/IEC 7816-4	Write data to a data file (EF)
	ReadData	Proprietary	Read data from a data file (EF)
	WriteData	Proprietary	Write data to a data file (EF)
Authentication	AuthenticateEV2First	Proprietary	Perform AES three-pass authentication
	AuthenticateEV2NonFirst	Proprietary	Perform AES three-pass authentication
	AuthenticateLRPFirst	Proprietary Proprietary	Perform LRP three-pass authentication
	AuthenticateLRPNonFirst		Perform LRP three-pass authentication
Key management	GetKeyVersion	Proprietary	Get version of the specified key
	ChangeKey	Proprietary	Update key, version and reset counters
Digital signature	Read_Sig	Proprietary	Get the tag digital signature
Metadata management	GetVersion	Proprietary	Get tag metadata (UID, producer)
	GetCardUID	Proprietary	Get the unique 7-byte tag UID
	GetFileCounters	Proprietary	Get the SDM read counter
	GetFileSettings	Proprietary	Get file metadata (access rights, SDM)
	ChangeFileSettings	Proprietary	Set file metadata (access rights, SDM)
	SetConfiguration	Proprietary	Set tag mode (LRP, random ID)

MF: master file; DF: dedicated file; EF: elementary file; LRP: leakage-resilient primitive; SDM: secure dynamic messaging; UID: unique tag identifier; AES: advanced encryption standard

Table 3. Comparison of three NXP tags designed for the IoT domain

Tag type	NDEF	Secure messaging	SDM	Random ID	Digital Sig.	Authentication	Memory access protection
NT4H2421Gx	√	√	√	√	√	√	√
NTAG21x	√	×	×	×	√	×	√
NTAG210×	×	×	×	×	√	×	×

IoT: internet of things; SDM: secure dynamic messaging

NTAG21x is protected by the same digital signature principle, while it relies on a different, weaker elliptic curve, which produces a 32-byte signature. NDEF and memory access protection are provided as well, while, for the latter, access is granted on a 32-bit password basis instead of the more reliable AES-128 authentication. The other features are not provided by this tag.

Concerning the last type, *NTAG210μ* does not provide any of the listed features, apart from the 32-byte digital signature.

Finally, none of the tags provides strong protection against chip cloning attacks. Concerning *NT4H2421Gx*, while a cloning attempt is not straightforward, since it implies that the malicious party needs to learn the AES originality keys, it is not impossible. Further considerations on the subject are provided in the "Discussion" section.

3 RESULTS

To effectively check the tag properties and some of its core functionalities, we designed a mobile application on the basis of Android OS, which uses the NFC sensor of the smartphone as a tag reader. The customized *NT4H2421Gx* tag was provided by *lab51 srl*.

In this section, we exemplified some of the APDU commands executed by the mobile application, and we stressed the digital signature verification process, as it represents the more reliable feature in relation to anti-forging. In the following, the content of each command and each response is proposed in hexadecimal format.

First of all, DF was selected through the standard *ISOSelectFile* command (see Table 2 for reference). Subsequently, the *GetVersion* command was addressed to acquire some basic information on the tag

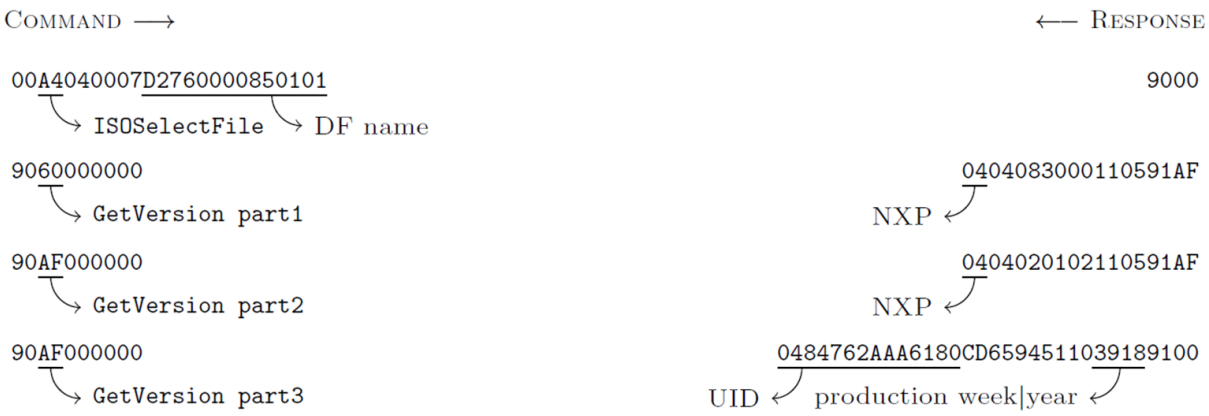


Figure 3. The complete communication trace concerning the *GetVerison* command. UID: unique tag identifier; DF: dedicated file

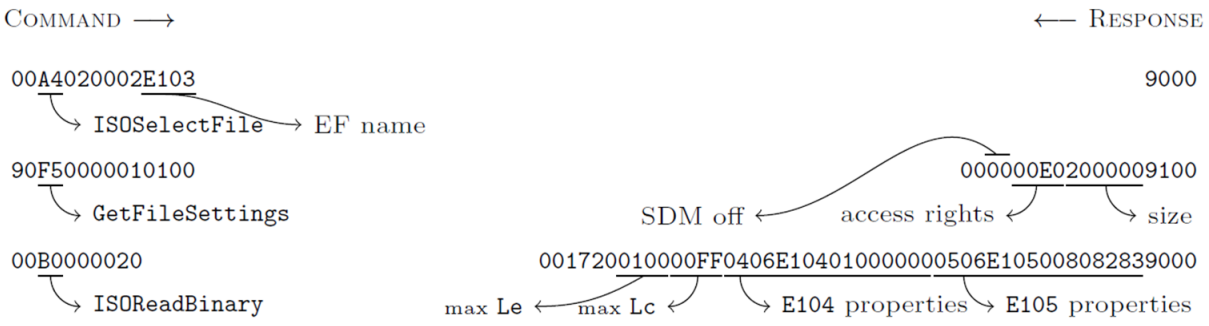


Figure 4. The complete communication trace concerning the commands performed against the *capability container* file. EF: elementary file; SDM: secure dynamic messaging

studied. The complete communication trace is provided in Figure 3. According to the returned data, the tag was produced during the 39th week of 2018 by NXP. The most important information included in the answer is the tag ID: as the tag studied is not configured with the *random ID* setting, the third response includes the real 7-byte UID. This condition may lead to a privacy breach and will be further discussed in the "Discussion" section.

The following step consists in the selection of the CC file. The application checks the file settings through the *GetFileSettings* command and subsequently reads the full file content using the standard *ISOReadBinary* command. The communication trace involved is provided in Figure 4. The information returned by the *GetFileSettings* command shows that the SDM is not enabled for this file. Again, the CC file has a size of *20:00:00*, which means it is composed of 32 bytes, as it should be interpreted with least significant byte encoding. Concerning the access rights to the file, the response shows that the *E103* file is subject to the *00:E0* access policy. According to NXP specifications, it means that this file is free to read (*E*), while other operations (write and change file permissions) need to be preceded by authentication through the key number *0x00* (the App Master Key). *ISOReadBinary* asks the tag for 32 bytes from the aforementioned file. The answer states that the CC effectively occupies 23 bytes only (*00:17*). Here, we may see that the file system comprises two more files, named *E104* and *E105*. The first one occupies 256 bytes and may be read and written without any authentication (*00:00*). Note that this access notation differs from the one returned by the *GetFileSettings* command as it is intended to be in accordance with the NFC Forum specifications. The latter file occupies 128 bytes. The access conditions for this file are set to *82:83*. These numbers fall in the proprietary range, concerning NFC Forum access policies. Specifically, it means that read operations need to be preceded by authentication with the application key number *0x02*. The same applies to write operations, with key number *0x03*.

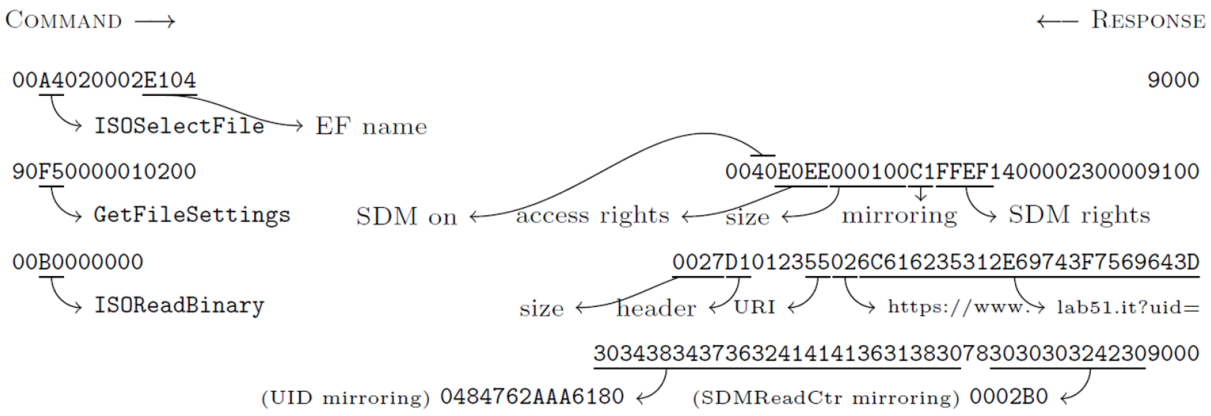


Figure 5. The complete communication trace concerning the commands performed against the NDEF file. EF: elementary file; SDM: secure dynamic messaging; UID: unique tag identifier

The next step consists of the inspection of the NDEF file. After the file selection, the application checks the file settings through the *GetFileSettings* command and, subsequently, reads the full file content using the standard *ISOReadBinary* command. The complete communication trace is provided in Figure 5. The information returned by the *GetFileSettings* command shows that, differently from the CC file, SDM is enabled for this file. Specifically, the file metadata shows that two attributes are supposed to be mirrored inside the NDEF file: the tag UID, stored at offset *14:00:00* (i.e., 20 following the decimal notation) and the SDM read counter, stored at offset *23:00:00* (i.e., 35 following the decimal notation). Both of them are stored in ASCII encoding. SDM access rights are set to *FF:EF*; this means that the UID and the *SDMReadCtr* are stored as plaintext within the NDEF file. Moreover, no run time encryption is applied to these data when the NDEF file is read through the *ISOReadBinary* or *ReadData* commands. Again, the *GetFileCounters* command is disabled. Moreover, metadata indicate that the overall dimension of the NDEF file is 256 bytes (*00:01:00*), and the access conditions are set to *E0:EE*. This access policy reflects the one included in the CC file for the NDEF file, as it states that the file may be updated and read with no restrictions (*E*). This setting suggests that the default file access rights were not changed after chip personalization.

Concerning the file content, the file effectively occupies 39 bytes (*00:27*). The file stores a single NDEF record having header *D1*. Hence, this record is a *short record* of a *well-known type*. The specific type is a URI (*55*) and the payload length is 35 (*23*). The first byte of the URI is *02*, which is an abbreviation for "*https://www.*". The remaining bytes contain the rest of the URI and the mirrored UID and *SDMReadCtr*, stored in ASCII encoding, as depicted in Figure 5.

To check the correctness of the APDU implementation in relation to the tag access logic, we also tested two more commands: *GetCardUID* and *GetFileCounters*. Both commands correctly return an error code. In the first case, this is due to the fact that the command was executed when the tag and the reader were not under authenticated mode. The error returned by the latter is instead related to the *SDM access rights* reported in Figure 5: as the *SDMCtrRet* is set to *F*, the *GetFileCounters* command is disabled. The error codes are provided in Figure 6.

Finally, we run the *Read_Sig* command to verify the digital signature and to prove the compliance of this tag with respect to chip forging. The related communication trace is listed in Figure 7.

According to NXP, the digital signature relies on elliptic curve cryptography[25] and was produced for the tag UID using an *ECDSA* algorithm with the elliptic curve *secp224r1*. As the name suggests, this curve implies keys of 224 bits (i.e., 28 bytes). Thus, the digital signature is composed of two parts: the first part is

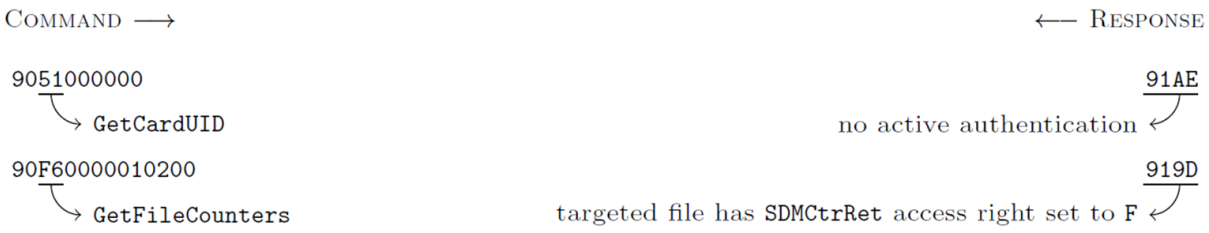


Figure 6. The complete communication trace concerning the *GetCardUID* and *GetFileCounters* commands

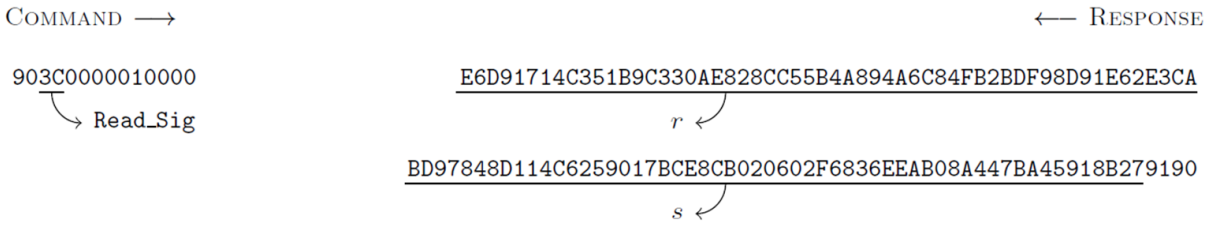


Figure 7. The complete communication trace concerning the *Read_Sig* command

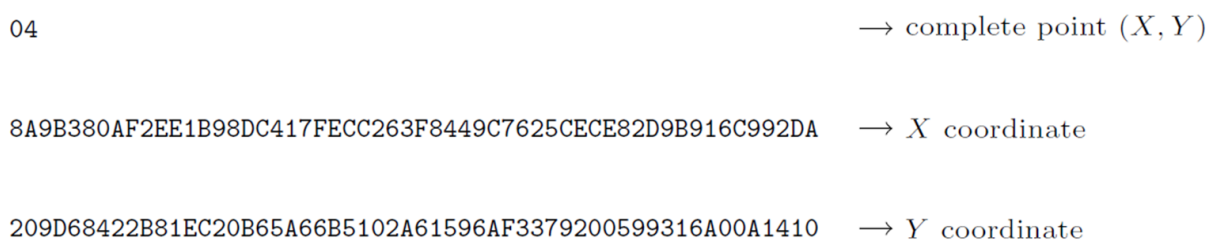


Figure 8. NXP public key for the elliptic curve *secp224r1*

28 bytes long and refers to the *r* parameter, and the second part is 28 bytes long as well and refers to the *s* parameter. The corresponding *public key* which should be used to verify the digital signature is provided by NXP and includes the *X* and *Y* coordinates of a point on the curve, plus an additional control byte[26]. The public key is provided in Figure 8.

The verification procedure was written within the Android application relying on the *Bouncy Castle Cryptographic Library* (https://www.bouncycastle.org).

To correctly test the digital signature, the raw bytes returned by the *Read_Sig* command need to be encoded in DER; otherwise, they cannot be handled by the java library used to operate the verification.

The verification procedure may be summed up as follows:
1. add the *Bouncy Castle* security provider;
2. create an empty data structure based on the *secp224r1* curve;
3. load the elliptic curve point from the raw bytes containing the NXP public key;
4. generate the elliptic curve public key accordingly;
5. prepare a *Signature* object with the aforementioned public key;
6. set the message to be verified as the tag UID;
7. encode the tag digital signature with DER encoding;
8. perform the digital signature verification on the DER-encoded signature.

The Android algorithm correctly verifies the digital signature. The originality check based on strong asymmetric cryptography is thus passed.

4 DISCUSSION

In this section, we discuss some notable security and privacy patterns that may be addressed using the *NT4H2421Gx* tag.

4.1 Communication channel security

The most commonly known security functionalities are based on three-pass mutual authentication and rely on AES symmetric cryptography. The authentication procedure is initiated by the *AuthenticateEV2First* or *AuthenticateLRPFirst* command.

When the reader and the tag are in the authenticated state, they are able to communicate using each command included in the command set. Performing a successful authentication proves that the reader possesses one of the cryptographic keys listed in Table 1. In authenticated mode, each APDU command is protected by *secure messaging*. Thus, message payloads are encrypted using the adopted AES key, and a *message authentication code* is attached as well. It follows that the communication channel is secured with respect to sniffing/eavesdropping attacks. Three-pass mutual authentication and secure messaging ensure confidentiality, integrity and trust. Of course, as they rely on symmetric AES cryptography, they suffer the key distribution problem, which is notably relevant within this field[27]. Some strategies should be adopted to provide the readers with one or more AES keys.

Finally, the *SetConfiguration* command may be used to enable the leakage-resilient primitive (LRP) mode (note that it is not possible to revert the tag to simple AES mode). Under LRP mode, three-pass authentication is started by the *AuthenticateLRPFirst* command and may rely on *originality keys* as well. LRP mode relies on a slightly different AES algorithm which is designed to resist side-channel attacks. An in-depth discussion on this subject falls out of the scope of this work.

4.2 Privacy implications

The GDPR specifically includes the term *online identifiers* within the definition of what constitutes personal data. These objects may include information relating to the device that an individual is using, such as applications, tools or protocols. To this end, the GDPR *Recital 30* shows a shortlist as an example and explicitly includes *RFID tags*. To comply with the latest privacy requirements, a good tag should thus be allowed to hide its UID under specific circumstances, since this UID may be sniffed out by unauthorized readers, threatening the user's privacy.

The *random ID* feature provided by *NT4H2421Gx* implements this requirement. This setting may be triggered through the *SetConfiguration* command, and prevent the UID to be unveiled through the *GetVersion* command. Specifically, when the tag is in *random ID* mode, a 4-byte random ID substitutes the 7-byte UID within the *GetVersion* response. There are two more options to learn the tag UID: using the *GetCardUID* command or reading it out from the NDEF file when UID mirroring is active. The first option does not represent a privacy breach, as the *GetCardUID* command is not allowed when the reader is not authenticated (see Figure 6 for reference). Concerning the latter option, it should be pointed out that UID mirroring is not mandatory, and moreover, the mirrored UID may be stored as ciphertext within the NDEF file. While the examined tag mirrors the UID as plaintext (see Figure 5 for reference), a proper change in the NDEF file settings (through the *ChangeFileSettings*) would encrypt the UID.

4.3 Chip cloning

The ability of a malicious stakeholder to clone the tag is probably the most dangerous event concerning *NT4H2421Gx*. The only countermeasure provided by the tag is represented by the inability of the attacker to copy the four *originality keys* which are stored in the ROM. These keys may be used accessing the MF level to perform a successful authentication, proving the tag originality. Unfortunately, it is evident from

NXP documentation that these symmetric AES keys are sometimes shared with NXP's licensees to check if the tags are genuine[26]. This information could be maliciously used to produce a complete clone of a genuine NXP tag.

To this end, please note that the tag UID and the corresponding NXP digital signature may be acquired through a legitimate tag inspection (as described in the "Results" section) and copied to the cloned tag as well.

To overcome this issue, further security protocols should be adopted. A significant example is represented by electronic passports[15]. The guidelines for e-Passport issuance and management are provided by the International Civil Aviation Organization (ICAO), and include a detailed description of the security protocols and the logical data structure used to store and arrange data into the RFID chip. To prevent chip cloning attacks, ICAO designed the *Active Authentication* security protocol. This protocol relies on asymmetric cryptography and requires a dedicated key pair. Briefly, during the chip's customization phase, the secret key is stored in the chip's secure memory, while the public key is stored in one of the chip's elementary files. When the reader needs to check whether or not the chip is genuine, it sends a random nonce to the chip, which signs it using the private key as signing key, according to the adopted cypher. The reader then reads the chip's public key from the corresponding EF and decrypts the string. On a positive match, the protocol succeeds. As the private key is stored in the chip's secure memory, it is very hard to read for an attacker. Moreover, as the protocol relies on asymmetric cryptography, there is no need for the licensees to handle the private key. This missing piece (the private key) and the introduced protocol represent a strong defense against chip cloning attacks. A similar solution could be adopted to strengthen the security features of *NT4H2421Gx*.

4.4 Tag forging

When we talk about tag forging, we refer to the ability of an attacker to produce a new tag from scratch claiming that it is genuine and that it is produced by some trusted organization (such as NXP). This procedure differs from the cloning one, as in this case the attacker does not copy the same tag UID in the forged chip, where the aim is to couple the tag with a new different UID.

The deep inspection performed on the *NT4H2421Gx* tag proved that this technology is strongly resistant with respect to forging activities. In fact, the *Read_Sig* command provides the reader with a digital signature which was computed signing the tag UID with an NXP elliptic curve *private key* (see Figure 7 for reference). Hence, to forge the tag, the attacker should sign the new UID with the same private key and should store the resulting signature in the tag ROM. Differently from symmetric AES keys, this private key never leaves the NXP hardware security module. As such, to forge a genuine NXP chip, the attacker must be able to break strong asymmetric encryption (which is usually deemed impossible under reasonable settings).

4.5 Soft security settings

To facilitate user experience and tag interoperability, this tag also supports a soft security setting named SDM. This feature may be set up for a single file (namely the NDEF one) through the *ChangeFileSettings* command. Besides, as depicted in Figure 5, SDM is enabled in the tag studied. SDM allows for confidential and integrity-protected data exchange, without requiring a preceding authentication. The NDEF file content may be accessed without any authentication. Encrypting part of the file content (together with tag UID or *SDMReadCtr*) is a valid option to reach the maximum interoperability with any RFID/NFC reader, while preserving some form of security. As predictable, when the involved application context requires strong security settings, SDM should not be considered a valid option.

This work could be extended according to several directions. From a theoretical point of view, a formal validation of the experimental results presented in this article would be an interesting open issue. Furthermore, a future research direction could involve further investigation of which countermeasures may be set up in this chip to handle chip cloning attacks better. Following the ICAO principles designed for electronic machine readable travel documents, a viable solution could consist of a novel protocol relying on asymmetric cryptography. Furthermore, this tag supports notable features that enhance privacy and also implement soft security settings, which increase tag interoperability. From a practical and application point of view, a good option could be to design and implement stateless systems (from the user's perspective) able to preserve some form of security and confidentiality while enabling tag inspection. Such a system could rely on smartphones NFC sensors and should be independent of a dedicated end-user application on the smartphone itself. This setting should exploit the SDM feature provided by the tag.

In a conclusion, in this paper we investigated the capabilities of the *NT4H2421Gx* tag. To effectively check the tag properties and some of its core functionalities, we designed a mobile application based on *Android OS* which uses the NFC sensor of the smartphone as a tag reader. This application allowed us to read the memory of the aforementioned chip at the bit level, and to discuss its core functionalities and settings in relation to the most common security and privacy patterns. In the final part of the paper we considered each of these aspects separately to stimulate the research community regarding these topics. Concluding, the deep Android inspection performed on the *NT4H2421Gx* tag showed that it represents an option to rely on when we need to design secure IoT applications. This tag is resistant to forging activities, and it also preserves confidentiality and authenticity on exchanged data. Again, SDM and mirroring enable stateless applications (from the user's perspective) to be delivered and also allow the surpassing of several privacy limitations.

DECLARATIONS

Authors' contributions

Made substantial contributions to conception and design of the study and performed data analysis and interpretation: Calderoni L
Provided technical and material support: Spadazzi L
Supervised the work: Maio D, Margara L

Availability of data and materials

Not applicable.

Financial support and sponsorship

None.

Conflicts of interest

All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

REFERENCES

1. Conti M, Dehghantanha A, Franke K, Watson S. Internet of things security and forensics: challenges and opportunities. Future Generation Comp Syst 2018;78:544-6.
2. Palmieri P, Calderoni L, Maio D. Private inter-network routing for wireless sensor networks and the internet of things. Proceedings of the Computing Frontiers Conference (CF'17). ACM, New York, USA; 2017;396-401.
3. Mehta R, Sahni J, Khanna K. Internet of things: vision, applications and challenges. Procedia Computer Sci 2018;132:1263-9.
4. Jia X, Feng Q, Fan T, Lei Q. RFID technology and its applications in Internet of Things (IoT). Proceedings of the 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet), Yichang; 2012:1282-5.
5. Lee YM, Cheng F. Exploring the impact of RFID on supply chain dynamics. Proceedings of the 36th conference on Winter simulation; 2004 Dec 5-8; Washington, DC, USA; 2004. pp. 1145-52.
6. Wu L, Liu S, Zhao B, Wu W, Zhu B. The research of the application of the binary search algorithm of RFID system in the supermarket shopping information identification. J Wireless Com Network 2019;27:1-10.
7. International Civil Aviation Organization. Doc 9303 - Machine readable travel documents. 7th ed. ICAO; 2015.
8. Avoine G, Calderoni L, Delvaux J, Maio D, Palmieri P. Passengers information in public transport and privacy: can anonymous tickets prevent tracking? Int J Information Management 2014;34:682-8.
9. Chothia T, Smirnov V. A traceability attack against e-passports. In: Sion R, editor. Financial cryptography. Lecture notes in computer science. Springer; 2010. pp. 20-34.
10. Ma D, Saxena N, Xiang T, Zhu Y. Location-aware and safer cards: enhancing RFID security and privacy via location sensing. IEEE Trans Distributed Secure Computing 2013;10:57-69.
11. Yang SJ, Huang X. Certain types of M-fuzzifying matroids: a fundamental look at the security protocols in RFID and IoT. Future Generation Computer Systems 2018;86:582-90.
12. Halevi T, Li H, Ma D, Saxena N, Voris J, et al. Context-aware defenses to RFID unauthorized reading and relay attacks. IEEE Transactions Emerging Topics Computing 2013;1:307-18.
13. He D, Zeadally S. An analysis of RFID authentication schemes for internet of things in healthcare environment using elliptic curve cryptography. IEEE Int Things J 2015;2:72-83.
14. Li N, Mu Y, Susilo W, Guo F, Varadharajan V. Vulnerabilities of an ECC-based RFID authentication scheme. Security Comm Networks 2015;8:3262-70.
15. Calderoni L, Maio D. Cloning and tampering threats in e-passports. Expert Syst Appl 2014;41:5066-70.
16. Gandino F, Montrucchio B, Rebaudengo M. Tampering in RFID: a survey on risks and defenses. Mobile Netw Appl 2010;15:502-16.
17. Gao L, Zhang L, Lin F, Ma M. Secure RFID authentication schemes based on security analysis and improvements of the USI protocol. IEEE Access 2019;17:1360-6.
18. Aghili SF, Mala H, Kaliyar P, Conti M. SecLAP: secure and lightweight RFID authentication protocol for Medical IoT. Future Gener Comput Syst 2019;101:621-34.
19. NXP Semiconductors: NT4H2421Gx - NTAG 424 DNA. Tech. rep., NXP (January 2019), rev. 3.0. Available from https://www.nxp.com/docs/en/data-sheet/NT4H2421Gx.pdf [Last accessed on 3 Jun 2020]
20. International Organization for Standardization Electrotechnical Commission. Organization, Security and Commands for Interchange. 3th ed. 2013.
21. Finkenzeller K, Muller D. RFID handbook: fundamentals and applications in contactless smart cards, radio frequency identification and near-field communication. 3th ed. Wiley; 2010.
22. NFC Forum: Type 4 Tag Operation Specification. 2011. Available from https://nfc-forum.org/product/nfc-forum-type-4-tag-specification-version-1-1/ [Last accessed on 3 Jun 2020]
23. NFC Forum: NFC Data Exchange Format (NDEF). 2006. Available from https://nfc-forum.org/product/nfc-data-exchange-format-ndef-technical-specification/ [Last accessed on 3 Jun 2020]
24. Daemen J, Rijmen V. The Design of Rijndael: AES - The Advanced Encryption Standard. Information Security and Cryptography, Springer; 2002.
25. Miller VS. Use of elliptic curves in cryptography. In: Williams HC, editor. Advances in Cryptology - CRYPTO'85, Proceedings. CRYPTO 1985. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg; 1985;218:417-26.
26. NXP Semiconductors: AN12196 - NTAG 424 DNA and NTAG 424 DNA TagTamper features and hints. Tech. rep., NXP (July 2019), rev. 1.5. Available from https://www.nxp.com/docs/en/application-note/AN12196.pdf [Last accessed on 3 Jun 2020]
27. Ng CY, Susilo W, Mu Y, Safavi-Naini R. Practical RFID ownership transfer scheme. J Computer Security 2011;19:319-41.

Clédel *et al. J Surveill Secur Saf* 2020;1:119-39
DOI: 10.20517/jsss.2020.08

Journal of Surveillance, Security and Safety

Original Article

Open Access

Resilience properties and metrics: how far have we gone?

Thomas Clédel[1], Nora Cuppens[1,2], Frédéric Cuppens[1,2], Romain Dagnas[3]

[1]Department of systems, networks, cybersecurity and digital law, IMT Atlantique, Cesson-Sévigné 35510, France.
[2]Department of IT and software engineering, Polytechnique Montréal, Montréal, QC H3T 1J4, Canada.
[3]Cybersecurity Team, IRT SystemX, Palaiseau 91120, France.

Correspondence to: Prof. Nora Cuppens, Department of IT and software engineering, Polytechnique Montréal, 2500 Chemin de Polytechnique, Montréal, QC H3T 1J4, Canada. E-mail: nora.boulahia-cuppens@polymtl.ca; ORCID: 0000-0001-8792-0413.

How to cite this article: Clédel T, Cuppens N, Cuppens F,Dagnas R. Resilience properties and metrics: how far have we gone?. *J Surveill Secur Saf* 2020;1:119-39. http://dx.doi.org/10.20517/jsss.2020.08

Received: 6 Mar 2020 **First Decision:** **Revised:** 31 Aug 2020 **Accepted:** 31 Oct 2020 **Published:** 30 Nov 2020

Academic Editor: Xiaofeng Chen **Copy Editor:** Cai-Hong Wang **Production Editor:** Jing Yu

Abstract

Aim: Resilience is discussed among researchers and practitioners for several decades, but its definition has been questioned even recently and many methods are proposed to evaluate the resilience of systems. This paper presents a review of historic and recent research articles that define and/or propose a way to measure resilience of systems.

Methods: While definitions are classified according to the ideas they focus on, different categories of metrics are described, such as quantitative or qualitative approaches.

Results: This paper points out that many metrics tend to valuate resilience similarly. In fact, they are generally built upon a specific definition. On the other hand metrics can also be really heterogeneous and do not capture the same meaning of system resilience when different definitions of resilience are considered.

Conclusion: This paper aims at gathering and comparing metrics and definitions of resilience in order to determine the origins of the particularities and classify them according to the attributes they take into account.

Keywords: Resilience, metrics/measurement, survey

1. INTRODUCTION

Risk assessment has been the dominant paradigm for system design and management for decades, especially in the case of cyber-physical systems (CPS). These systems are used in critical infrastructures, and a "well-designed risk assessment of CPS will provide an overall view of CPS security status and support efficient allocations of safeguard resources"[1]. Furthermore, "With an understanding of risk, it is then possible for an operator to prioritise the implementation of resilience measures"[2] (additional research results related to this work are available at: http://www.cost-recodis.eu). However, unprecedented adverse events such as natural disasters (the Fukushima Daiichi nuclear accident) or cyber-attacks (StuxNet or BlackEnenergy) have caused unexpected losses. These events have highlighted some weaknesses of well-established models and frameworks. As a consequence, it has recently been accepted by scientific communities and governments that risks threatening critical infrastructure cannot all be identified or prevented and that there is a need for new approaches to mitigate damages. Resilience emerged from this lesson as the logical way to overcome the limitations of previous dominant approaches that are risk assessment and system safety.

While systems were considered safe by design and failures caused by human errors, it is now accepted that mismatches exist between administrative procedures and the ways in which systems actually run. Indeed, normal system performance, resulting from required adjustments, adaptations, and optimizations must be distinguished from normative system performance that is prescribed by rules and regulation[3].

Some studies and audits have been conducted in modern industries and different environments to assess whether resilience was considered during the design and planning phases of industrial processes, and how resilience strategies are applied during the operational phase. Studied environments include nuclear plants[4], electricity distribution[5], chemical plants[6,7], sea fishing[8], oil distribution plants[9], railways[10], *etc*. Carvalho *et al.*[4] introduced a framework for the analysis of micro incidents during nuclear power plant operations. Saurin *et al.*[5] improved a method for assessing health and safety management systems. Azadeh *et al.*[6] presented a new concept of resilience engineering, which includes teamwork, self-organization, redundancy, and fault-tolerance, while Shirali *et al.*[7] identified the challenges that occur in the process of building resilience engineering and its adaptive capacity in a chemical plant. Morel *et al.*[8] focused on "the relationship between resilience and safety, and discusses the choice of strategies for safety-improving interventions, taking into account the system's financial performance and the legal pressure to which it is subjected". Abech *et al.*[9] presented the challenges in order to improve resilience in an oil distribution plant. Hale *et al.*[10] proposed an evaluation, which shows that railways are "examples of poor, or at best mixed, resilience, which can, however, still achieve high levels of safety, at least in certain areas of their operations". Most of these studies conclude that some resilience mechanisms inherently exist in these environments. However, these resilience mechanisms may not always be recognized as such by employees. They demonstrate how people adapt to challenging situations where operational, plannings and procedures are in conflict.

The absence of consensus for a definition of resilience, as well as the abundance of metrics evaluating resilience and the over-dominance of risk assessment and system safety, can explain that resilience is rarely applied and considered as a system design and management paradigm. However, it can be noticed that definitions and metrics are not as heterogeneous since only few criteria are used in the current article to classify them. While some metrics clearly differ from the others and do not evaluate the same "resilience", many definitions and metrics are in fact variations of others. Some of them can be considered as refinements of older metrics or definitions. Occasionally, variations can be justified by a will to produce a domain specific evaluation of resilience.

The goal of this article is not to provide an exhaustive list of articles that deal with resilience. Many articles propose mechanisms, techniques, and technologies to improve resilience of systems but fewer articles provide their own definition and/or metric of resilience, and fewer still provide an original definition or metric. In fact,

many measures and definitions are derived from more original ones, so that those that share a common origin also share many characteristics. The current paper aims at gathering and comparing metrics and definitions of resilience so that common criteria and differentiation criteria emerge from them. This way, categories of metrics and definitions can be defined. To identify pertinent literature, online database searching was performed on databases such as Web of Science and DBLP. Articles were filtered with the keyword "resilience" and a set of other keywords, including "metrics", "measure", evaluation", and "framework". The most relevant were selected on the basis of their titles, abstracts, and whether they applied to the field of engineering. A second step in this research consisted in cross-referencing the sources of the previously selected articles in order to determine the origins of the particularities of their definitions and metrics.

This paper is organized as follows. Section 2 provides a survey of definitions of resilience, from its original definition in ecological system to recent definitions in networks and cyber-physical systems. Definitions are classified according to the ideas they focus on. Because there are many definitions for resilience, the expected attributes of a resilient system can slightly differ from one article to an other. Thus, a description of the various attributes associated with resilience is given in Section 3. Then, a survey of different metrics used to evaluate the resilience of systems is provided in Section 4. Some metrics consist in measuring separately some attributes of resilience and then combining them. Others evaluate resilience without considering what the various capacities that compose resilience, and they measure the impact of harmful events that occurred on a system to assess the level of resilience of this system for these events. All considered metrics are classified according to the attributes they take into account. The results of this classification are summarized in a table at the end of the section. Since resilience is a complex property, it may often be confused with other concepts and system properties. Section 5 provides results of some articles that compare resilience with other properties such as robustness and risk assessment. Section 6 discusses the existing limitations and gaps in the described definitions and metrics. Additionally, it provides the conclusion of this study.

2. RESILIENCE DEFINITIONS

The term "resilience" comes from the Latin word "resilire", which has several interpretations such as "to rebound", to "spring back", or "to withdraw into oneself". Even if the current meaning of "resilience" differs slightly from its Latin origin and despite the diversity of definitions, most of them fit with at least one of these antic meanings. The resilience perspective emerged in the 70s from ecology with the work by Holling[11]. A few years later, the resilience concept began to influence other fields such as anthropology, sociology, or psychology, as described in[12], before it reached engineering sciences and, even more recently, into computer science and information technologies.

The notion of resilience was first developed in some domains such as ecology with the work by Holling[11]. Resilience of a population is defined as a system property where the system behavior is less important than the system persistence. Thus, resilience is distinguished from stability. The author described it as the capacity of a system to move from a stability domain into another one and put the emphasis on "a high capability of absorbing periodic extremes of fluctuations", the maintainability of "flexibility above all else", and a capacity to "restore its ability to respond to subsequent unpredictable environmental changes". Historically, resilience has also been developed in psychology and refers to the ability to recover from trauma and crisis[13] while "childhood resilience is the phenomenon of positive adaptation despite significant life adversities"[14].

2.1. A system property

Francis and Bekera[15] described resilience as a system property to endure undesired events in order to ensure "the continuity of normal system function". This ability corresponds to three system's capacities: absorptive, adaptive, and restorative capacities. It could be considered that this definition goes against the original concept of resilience given by Holling[11] as the continuity of normal function can be considered as a synonym of system

stability. However, the authors also specified that resilience postulates flexibility in terms of performance, structure and function while these changes are not irreversible or unacceptable.

Resilience is also defined as the maintenance of "state awareness and an accepted level of operational normalcy in response to disturbances"[16]. Operational normalcy corresponds to the maintenance of "stability and integrity of core processes" according to McDonald[17] and resilience was described by Wreathall[18] as the ability to "keep, or recover quickly to, a stable state". These definitions confirm the previous description as resilience focuses on some operational stability even if systems are supposed to "tolerate fluctuations via their structure, design parameters, control structure and control parameters"[19]. A new point highlighted by this definition is the need to collect and fusion data concerning the current state of the system. This knowledge aims at knowing the current date of the system and its environment and is a basis for decisions[18]. Processes to collect, fuse, and prioritize information should be considered when designing resilient systems. Indeed, resilient systems should not be considered as a single technology but as a complex integrated system of systems that ensures coordination among subsystems through communication and sharing of information[20].

2.2. Resilience is related to service delivery

Sterbenz *et al.*[21] considered systems as networks, and their resilience is defined as the ability "to provide and maintain an acceptable level of service in face of various faults and challenges to normal operation". This definition is close to another one given by Laprie[22], where resilience is "the persistence of service delivery that can justifiably be trusted, when facing changes". For both definitions, resilience focuses on service delivery and particularly on avoidance of service failure. System services are the system behavior as it is perceived by its users[23]. They are different from system functions which correspond to the expected result of the system behavior, in other words what the system is intended to do. Delving into a more specific domain of cyber-physical system, Clark and Zonouz[24] defined resilience as the "maintenance of the core [. . .] set of crucial sub-functionalities despite adversarial misbehaviors" and a guarantee of "recovery of the normal operation of the affected sub-functionalities within a predefined cost-limit". Again, this definition reinforces the need to maintain a service delivery above a fixed threshold. If a perturbation leads the system to be under this threshold, then the system is in an unacceptable state and has failed to be resilient.

Power systems are also considered[25], and resilience is defined as the "ability to maintain continuous electricity flow to customers given a certain load prioritization scheme". According to the authors, traditional risk assessment is not the best approach to achieve resilience as resilience concerns "unexpected rare extreme failures" whose likelihood cannot be easily estimated. Thus, this definition completes the previous ones as it focuses on service delivery and underlines that some services are more critical than others and should not be interrupted.

2.3. Events handling

A commonly accepted definition of resilience was given by Vugrin *et al.*[26]. Resilience is described as the ability of a system, for a given disruptive event, to "reduce 'efficiently' both the magnitude and the duration of the deviation from targeted 'system performance' levels". This definition has frequently been used to propose resilience metrics based on system performance such as some metrics detailed in Sections 4.1 and 4.2. This definition and its derived metrics also imply that a system has different levels of resilience to different disruptions and an evaluation of resilience is needed for every specific disruption.

Ayyub's definition of resilience is close to the previous one[27], as resilience is said to be "the ability to prepare for and adapt to changing conditions and withstand and recover rapidly from disruptions". On the contrary of the previous definition, resilience is not only concerned with the occurrence of disruptions, but is also considered in a pre-disruption phase as a need for preparation and evolution is pointed out by this definition.

Another similar definition was given by Haimes[28] as resilience is "the ability of a system to withstand a major disruption within acceptable degradation parameters and to recover within an acceptable time and composite costs and risks". Compared to the previously described definitions, Haimes pointed at the need to estimate the cost of the recovery process.

Another definition of resilience was considered by Mauthe *et al.*[2]. This definition is applied to communication networks: "Resilience of a communication network is its ability to maintain the same level of functionality in the face of internal changes and external disturbances as a result of large-scale natural disasters and corresponding failures, weather-based disruptions, technology-related disasters, and malicious human activities."

However, some definitions do not consider the amplitude of disruptions. Dinh *et al.*[29] defined resilience as "the ability to recover as soon as possible after an unexpected situation". The authors nevertheless pointed out the need to minimize disruptions consequences but only with a view of faster recovery.

Hollnagel[3] defined resilience as "the ability of a system or an organization to react to and recover from disturbances at an early stage, with minimal effect on the dynamic stability." Hale and Heijer's definition describes resilience as "the characteristic of managing the organisation's activities to anticipate and circumvent threats to its existence and primary goals"[30]. Resilience is also "the ability of systems to prevent or adapt to changing conditions in order to maintain (control over) a system property"[31]. On the other hand, Sundström and Hollnagel described resilience as "an organizations ability to adjust successfully to the compounded impact of internal and external events over a significant time period"[32]. Another definition from Wreathall describes resilience as "the ability of an organization (system) to keep, or recover quickly to, a stable state, allowing it to continue operations during and after a major mishap or in the presence of continuous significant stresses"[18].

2.4 Other definitions

Recent work suggests looking at resilience with a different perspective. Thompson[33] considered a system as a set of resources for which particular states are expected, such as ensuring personal safety, preserving confidentiality of a database, etc. Security is the system capacity to maintain expected states of resources. However, security breaches can occur and resilience is defined as "the maintenance of a nominated state of security". This resilience is achieved by detecting, containing, and resolving a security breach. While many approaches only consider resilience of accidental faults, this one seems to focus only on attacks. We provide a classification of resilience definitions in Table 1

3. DESCRIPTION OF RESILIENT SYSTEMS

It is commonly accepted that resilience of a system is supported by three system capacities. These capacities were first described in 1973[11]. Holling compared the resilience of a population with a game "in which the only payoff is to stay in the game". Thus, a resilient population has "a high capability of absorbing periodic extremes of fluctuation", maintains "flexibility above all else", and can "restore its ability to respond to subsequent unpredictable environmental changes". They are known as absorbability, adaptability, and restorability and are considered so central to the notion of resilience that they are frequently used to define resilience[15,34]..

3.1. Absorbability

This capacity is "the degree to which a system can automatically absorb the impacts of systems perturbations and minimize consequences with little effort"[26]. Considering power systems, Arghandeh *et al.*[25] explained that the absorbing potential of a system "depends on the components" design characteristics, the system topology, the control philosophy, and the protection coordination". Indeed, features such as robustness, redundancy, diversity, and defense in-depth enhance the absorbability of a system and provide higher survivability[20]. This capacity is sometimes designed as buffering capacities[35] and corresponds to the maxi-

Table 1. Table of resilience definitions

Reference	Definition orientation				Goal
	Events handling	System stability	Service delivery	Resilience capacities	
Ayyub [27]	✓				Preparation, adaption, resistance, recovery
Dinh *et al.* [29]	✓				Fast post-event recovery
Haimes [28]	✓				Acceptable degradation, time, and costs
Vugrin*et al.* [26]	✓				Reduction of the performance level deviation
Werner [13]	✓				Psychological and social adaptation
Hollnagel [3]	✓				Recover from disturbances at an early stage
Hale and Heijer [30]	✓				Managing activities, anticipation of threats
Leveson *et al.* [31]	✓				Prevent/adapt to maintain a system property
Sundström and Hollnagel [32]	✓				Ability to adjust in a long time period
Wreathall [18]	✓				Continuity of operations during/after a mishap
Mauthe *et al.* [2]	✓				Same level of functionality in case of changes
McDonald [17]		✓			Stability and integrity of core processes
Rieger [16]		✓			State awareness and operational normalcy
Wreathall [18]		✓			Keeping or quick recovery of a stable state
Arghandeh *et al.* [25]			✓		Continuity of electricity flow
Clark and Zonouz [24]			✓		Service delivery and guarantee of recovery
Sterbenz *et al.* [21]			✓		Maintenance of an acceptable level of service
Thompson *et al.* [33]			✓		Maintenance of security state
Francis and Bekera [15]			✓	✓	Continuity of normal service function
Holling [11]				✓	Population survival
Wei and Ji [34]				✓	Incidents handling

mal amplitude of disruptions that can be tolerated. To buffering capacities, Woods specified a need for margin and tolerance assessments that determine how closely and how well a system is currently running near to its performance boundaries.

Moreover, resilience is not directly associated with a capacity to absorb and mitigate incidents [22,36]. However, a need for diversity is specified as it prevent vulnerabilities to become a single point of failure. This diversity manages the vulnerabilities of components to incidents by the use of different components and processes for similar functions, but it should also consider the exposition of components and processes to these incidents with geographic or topological dispersion for example. Dinh *et al.* [29] decomposed absorbability into two complementary properties. The first property is flexibility and can be considered as a synonym of stability in the cited article, as it consists in maintaining the system production variation into a desired range while inputs are changing slightly. The second property is controllability and indicates how easily a system can be brought in a desired state.

3.2. Adaptability

Adaptability [26], also known as flexibility [35], is "the degree to which the system is capable of self-reorganization for recovery of system performance" and is described as "the ability to replace component or input with another" or the "system's ability to restructure itself" to face changes and external pressures. While this description could be associated with diversity, which is more commonly interpreted as part of absorbability, adaptability is also concerned with changing the system structure, policies, and priorities to mitigate the impact of a disruption.

Some works refer to adaptability as evolvability [22,36]. It represents the ability of a system to "accommodate changes" by upgrading itself with new functions or technologies during design and implementation phases or by dynamically adjusting its behavior or its architecture to face operational faults and attacks. Moreover, in [30], the authors affirmed that resilience has to be continuously kept up-to-date as it can disappear or be ineffective against specific threats.

One possible adaptive mechanism is the use of safe mode controls. It consists in using simple but extremely reliable systems that prevent critical failures [20]. Safe mode depends on few input sources such as Earth's

magnet field is used to control spacecraft stability[37], and the used sensors are reliable and redundant enough so that the safe mode system is considered "fail safe". By definition, safe mode is designed to limit the impact of a perturbation but not to mitigate it. It ensures a minimal system function.

3.3. Recoverability

Recoverability is determined by internal and external entities and their capacity to easily restore the system to its original state or a better one. It consists in dynamic mechanisms such as repairing or replacing damaged components, reinitializing components to a proper state, etc. While adaptability can alter the system structure to preserve or restore system performance, recoverability aims at "returning a system to near its original structure"[26]. Moreover, adaptive changes are in general temporary, whereas restorative changes are expected to be as permanent as possible.

3.4. Other capacities and descriptions

While the works[22,36] described absorbability (with diversity) and adaptability (evolvability) as resilience capacities, restorability is not considered. In place of it, it is claimed that a resilient system has "assessability" and usability. Assessability is the ability to verify and evaluate if a system behaves properly and if the quality of service is delivered. This verification and evaluation can be performed during design and pre-deployment phases but should also be an ongoing process as systems are supposed to evolve. Usability describes how ergonomic user interfaces are. It consists in measuring how easy it is to learn basic tasks, memorize them, and avoid errors; how quickly tasks can be performed; and how pleasant the interface is to use. Usability is needed as systems are more and more complex and errors can lead to critical failures.

Some works[29,34] describe a resilient system as one that can anticipate and handle unexpected events. They describe capacities that such systems have: security (minimization of the incidence of undesirable events), mitigation/minimization capacity, and recovery ability. This description of resilience differs from the others for two reasons. Firstly, security is taken into account while resilience is generally considered only when an incident occurs, in other words, after security has failed. The second reason is the absence of adaptability amongst resilience capacities, even if the authors of both articles gave an example of minimization capacity that could be interpreted as adaptability. Indeed, minimization capacity includes an ability to detect disruptions and faults as soon as possible and to enable mitigation measures.

Resilience has been decomposed into three capacities[33]. First, a system must recognize and identify security breaches, which is a detection ability. A second capacity, containment, is the ability of a system to absorb and limit the impact of security breaches. The third capacity is resolution and consists in eradicating security breaches and restoring the system. Even if those capacities are not explicitly the three traditional ones, they are not unrelated. Recoverability is included in the resolution capacity. Detection and containment capacities have the same objectives as absorbability and adaptability: to maintain an acceptable level of service while facing and eradicating the security breaches. Although the authors did not describe how a system could face a security breach when detected, they pointed out that two resilience mechanisms come into play: survivability and impact limitation.

4.HOW TO MEASURE RESILIENCE

4.1 Quantitative deterministic

The articles described in this section use different measures for system performances or about some characteristics of an undesired event to build a metric of resilience. While most of these metrics provide a resilience value for a system, others consist in providing a score for different factors that compose resilience. They are denoted semi-quantitative approaches. The provided scores give clues concerning the resilience of a system but do not precisely result in a measure of it.

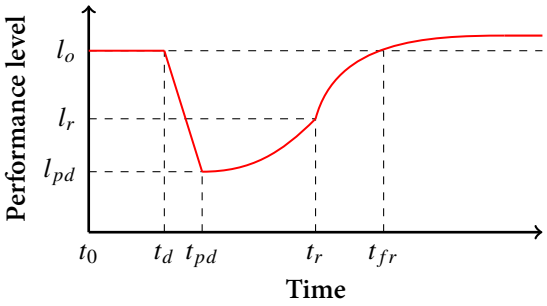


Figure 1. Performance level during the handling of a disruption (fault or attack).

Accidents and incidents cannot be considered as an absolute and direct indicator of system resilience [4]. External factors such as disturbances and attacks are not intrinsic properties of system resilience and their involvement in resilience metrics can be argued [38]. However, clues and markers of resilience can be provided by the analysis of the system dynamics and the interplay of its subsystems during the occurrence of these events.

With this in mind, several metrics evaluate resilience from the actual level of performance of a system during the occurrence of an unexpected event. Level performance can be used to illustrate different business cases [39] such as production capacity, quality, waste, cost, etc. The less performance is affected, the more resilient the system is. These metrics are event specific, which means that an event (fault or attack), or a set of events, is determined and the system resilience to this event is evaluated. It implies that resilience of a system should be evaluated for every known event or set of events that can occur in the system. This kind of metric is illustrated in Figure 1. Four times are generally considered. (1) t_d corresponds to the occurrence of a disruption. Before t_d, the system works at its original performance level l_o. (2) Despite absorption and adaptation mechanisms, the performance level is degraded by the disruption and reaches its lowest level l_{pd}. This moment is called the post-disruption time, t_{pd}. (3) Resilient mechanisms allow the system to partially recover until the disruption is resolved at time t_r. (4) Recovery mechanisms come into play and the system returns to its original level performance. The system has fully recovered from the disruption at t_{fr} but evolving capacities can allow the system to improve its performance after that.

The authors of [26,34] evaluated the performance loss due to a disruption as the integral of the difference between the original level and the actual level of performance on the interval $[t_d, t_{rf}]$. For the sake of comparison, Gholami *et al.* [40] proposed to use a per-unitized metric such that resilience is a ratio bounded in the range $[0, 1]$. Ayyub [27] proposed something similar but the expected performance level of the system is not constant over time; it decreases with aging effects. As a consequence, the older a system is before a disruption, the less resilient it is, as described below. Let $\mathcal{P}$ and $\mathcal{P}_{exp}$ be the time-dependent functions that correspond to the actual and expected performance levels of the system, respectively:

- Performance loss [34]:

$$\mathcal{P}_{loss} = \int_{t_d}^{t_{fr}} \left(l_o - \mathcal{P}\left(t\right)\right)\, dt \tag{1}$$

- Resilience ratio [40]:

$$\mathcal{R}_r = \mathcal{P}_{loss} \Big/ \int_{t_d}^{t_{fr}} l_o\, dt \tag{2}$$

[27]:

$$\mathcal{R}_r = \frac{td + F.\left(t_{pd} - t_d\right) + R.\left(t_{fr} - t_{pd}\right)}{t_{fr}} \tag{3}$$

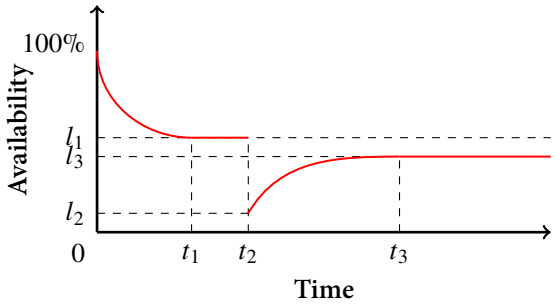


Figure 2. Availability of a system before, during, and after a shock[38].

with the failure profile

$$F = \int_{t_{pd}}^{t_d} \mathcal{P}(t)\ dt \Big/ \int_{t_{pd}}^{t_d} \mathcal{P}_{exp}(t)\ dt \tag{4}$$

and the recovery profile

$$R = \int_{t_{fr}}^{t_{pd}} \mathcal{P}(t)\ dt \Big/ \int_{t_{fr}}^{t_{pd}} \mathcal{P}_{exp}(t)\ dt \tag{5}$$

To this performance loss, called systemic impact[26], the authors added a recovery cost. This recovery cost corresponds to resources expended in recovery efforts, and, once combined with the performance loss, it gives the total loss due to a determined disruption, called recovery-dependent resilience[26].

Babiceanu and Seker[41] evaluated separately the loss of performance in three phases: degradation of performance from t_d to t_{pd}, balanced degradation from t_{pd} to t_r, and recovery of performance from t_r to t_{rf}. The evaluation is the same as the previous one: the integral of the difference between the original level and the actual level of performance over a period.

The resilience of a system to an event is evaluated by a resilience factor that is the product of three elements[15]: a degradation ratio l_{pd}/l_o, a partial recovery ratio l_r/l_o, and a speed factor t_r/t_δ. t_δ corresponds to the maximum acceptable value for t_r and $t_r > t_\delta$ implies that the system cannot recover from the disruption.

Cai *et al.*[38] used system availability instead of performance level. They defined availability as the ability to be in a state of performing a function if required external resources are provided. This approach is similar to the previously described ones in[15,26,41] and is depicted in Figure 2. The system begins at 100% of availability and then progressively reaches a stable level l_1 at time t_1. Then, n shocks impact the system at time t_2 and availability falls from l_1 to l_2. Resilience mechanisms handle these shocks such that availability reaches a post-shock steady state l_3 at time t_3. Thus, resilience is measured as the product of availability before and after shocks:

$$\text{(resilience)}^{[38]} : \mathcal{R} = \frac{l_1}{n \ln(t_1)} \sum_{i=1}^{n} \frac{l_3^i . l_2^i}{\ln\left(t_3^i - t_2^i\right)} \tag{6}$$

The authors claimed that the natural logarithm function is used to balance the availability and the recovery process of the system.

Sterbenz *et al.* proposed another approach to evaluate network resilience[42]. A system is composed of several layers: physical, link, topology, network path, end-to-end transport, and application. Each layer is represented

at a given time t by an operational state that consists in a $l \times m$ matrix of l operational metrics and m possible values, and a service state that consists in another $l \times m$ matrix of l service parameters and m possible values. Layers are overlapping such that the service state of a layer at time t becomes the operational state of the layer above at time $t + 1$. According to this model, the system resilience is evaluated at the boundary between two layers as the transition trajectory to move from the state of a layer to the state of the layer above.

Clark and Sonouz[24] used a linear time-invariant model to represent a system and its adversarial impacts. They considered a set of safe states and a basin of attraction that is a set of states allowing the system to return to a safe state under certain conditions. From these definitions, a system is considered resilient to an adversarial event as long as it remains in a safe state or in a state included in a basin of attraction. Since attackers can either physically attack the system or compromise input signals or inject false data, impacts of an attack are modeled as modified input and state matrices. Once a system and an attack are modeled, it can be determined if the system is resilient to this attack. Nonetheless, resilience can be evaluated as the amplitude of adversarial event that must impact the system to pull it out of safe states and basins of attraction. This idea of an attraction basin can be found in the original article of Holling[11], as described in Section 4.2.

4.1.1. Semi-quantitative approach

Shirali *et al.*[43] used six previously described resilient factors[18]: management commitment, reporting culture, learning culture, awareness, preparedness, and flexibility. Employees of an industry are divided into several groups corresponding to process units and are given a questionnaire. After gathering the questionnaires, a score from one to five is given for each resilient factor and for each group of employees. From these scores, managers can identify weaknesses in some resilient factors for some specific groups of employees. Despite this, interconnections between the six resilient factors or between groups of employees are not considered in this approach.

4.2.Quantitative probabilistic

Probabilistic approaches relate resilience with uncertainties and thus they add a stochastic component to the resilience evaluation. For several of them, denoted as event specific, this is the resilience of a system to a determined event that is evaluated. Generally, the probabilities considered in a resilience evaluation come from the stochasticity of occurrence of undesired events.

Originally, Holling did not provide metrics and methods to evaluate resilience in his article about resilience and stability of ecological systems[11]. According to Holling, resilience is only concerned with populations extinctions and resilience is the ability of a population to move from a stable population state to another one. Thus two parameters must be considered to evaluate resilience: the probability that an incident moves the population outside a stable state and the distance between stable states that determines how harmful the incident must be to lead to extinction. However, Holling explained that such measures require an immense amount of knowledge about the system.

4.2.1. Event Specific

Haimes claimed that resilience of a system can be determined only once a threat scenario is determined[28,44]: "the question 'What is the resilience of cyberinfrastructure X ?' is unanswerable". According to other articles, resilience can be evaluated only once all possible undesired events are determined[34]. For example, in addition to a quantitative deterministic evaluation of resilience, Babiceanu and Seker[41] provided two probabilistic metrics. The first extra metric is the probability of occurrence of a disruptive event that is the product of three other probabilities, the probability of a system to be vulnerable, the probability to be attacked, and the conditional probability of security to be bypassed (the attack is successful). The second extra metric is the probability of the system to recover from this event. It depends on the availability of a resilience solution for

this event, the conditional probability of this solution to be activated and the conditional probability of the system to recover once resilience mechanisms are engaged.

Once all undesired events are determined, resilience of a system is the sum, for all these events, of the probability of occurrence of each event multiplied by a resilience factor[15]. The resilience factor is system specific and event specific, as described in Section 4.1. For this metric, resilience factors are weighted with a fragility function that corresponds to a probability function of system failure. This fragility function is also event specific. On top of that, probabilities of the occurrence of events is combined with an entropy factor that represents the uncertainty of these probability distributions.

Thompson *et al.*[33] presented resilience as the maintenance of a security level and resilience is achieved in three steps: detection, containment, and resolution. According to this description, a metric based on these three capacities is proposed[45]. For a determined security breach, a probability is assigned to each of these capacities and represents the probability that the breach is detected, contained, or resolved. The authors argued that three events can lead to the restoration of the expected security state: (1) the breach is detected, then contained, and finally resolved; (2) the breach is detected and resolved without containment; and (3) the breach is resolved without detection or containment. As these events are independent, resilience is the probability that one of these events occurs.

Dynamic Bayesian networks are used[46] to represent a system. The resilience of a system to a disruption is expressed as the joint probability of the occurrence of the disruption and of the three resilient capacities: the probability to absorb, adapt to, and recover from the disruption. The authors described a nuclear plant, Fukushima Daiichi, as a set of eleven components such as Process Control System, Cooling System, Sea Wall, etc. These components contribute to at least one of the three resilience capacities, and the contribution of a component to one capacity is represented by a failure probability. Thus, 1–3 failure probabilities can be associated to each component. Nevertheless, as components can be involved in more than one resilient capacity, the three resilient capacities are not independent and Bayesian Networks are used to model these dependencies. The result of the application of this model is the time-dependent probability function of the resilience of a system to a determined disruption.

4.3. Fuzzy models

Fuzzy sets are a generalization of conventional set theory that were introduced by Zadeh[47] as a mathematical as well as natural way to deal with problems in which the source of imprecision is the absence of sharply defined criteria. They play an important role in human thinking such as determining if someone is tall or if something belongs to the class of animals. For example, while dogs are clearly classified as animals, it is more ambiguous concerning bacteria, plankton, etc. The articles given in this section use fuzzy sets and membership functions to build metrics for resilience.

According to Francis and Bekera[15], resilience is a designed and engineered property of a system. Moreover, Muller[48] proposed to separately evaluate system architectures through attributes such as redundancy, adaptivity, robustness, etc, for which numerous metrics already exist. To accommodate differences amongst metrics, system architectures are thus represented with fuzzy membership functions associated with evaluated resilience attributes. Using these membership functions, resilience attributes are combined using fuzzy rules to obtain a measure of resilience from a resilience membership function. An example of fuzzy rule is:

IF *adaptability* is *moderate* AND *robustness* is *high* THEN *resilience* is *high*

To evaluate organizational resilience, Aleksic *et al.*[49] proposed to consider a system as a network of processes. Processes have many resilience potentials, divided into three categories: (1) internal factors such as quality, human factors, or planning strategies; (2) external factors that are external capacities and capabilities; and (3)

enabling resilience factors such as detection and emergency response. These potentials are represented by fuzzy attributes and are given a value defined within [0, 1]. Uncertainties' attributes, such as the relative importance of resilience potentials for a specific process, are also considered and are given a similar value. Then, values assigned to all these fuzzy attributes, resilience potentials, and uncertainties are combined using membership functions to produce an estimation of the system resilience .

Azadeh *et al.*[50] used nine resilient factors/potentials contributing to a complex system resilience. While six of them were described[18] and used by Shirali *et al*. in a semi-quantitative metric[43], the authors added three factors: teamwork, redundancy, and fault-tolerance. Because these nine factors depend on each other, fuzzy cognitive maps are used to represent their interconnections and evaluate their contribution to system resilience. Following Aleksic *et al.*[49], membership functions are associated with each factor in order to evaluate the system resilience.

Clédel *et al.*[51] provided a framework to compare the resilience potential of different systems or configurations of the same system. The described model and metric cannot be used to determine if a system is resilient to a specific threat but it is used to determine if a system has more resilience potential than another one. A system is represented as a network of components. Components are service users of their previous components in the network and service providers of their next components. Services are represented through a partially ordered set of attributes, called data dimensions. Components inputs are fuzzy values associated with some dimensions. A value assigned to a dimension corresponds to the likelihood of this dimension to be externally consistent[52,53]. The article shows how these fuzzy values can be aggregated and manipulated so that components output fuzzy values associated with a set of data dimensions. Resilience is evaluated as follows: some nodes are the system client and their input values are fuzzy values for some expected dimensions. These expected dimensions correspond to services expected to be provided by the system, and their corresponding values are the likelihood for these services to be provided.

4.4. Frameworks

Some articles do not provide metrics or methods to evaluate the current resilience of a system. In place, they propose methodologies, guidelines, and good practices that are to be followed to design, maintain, and enhance the resilience of a system.

A framework for resilience, based on PAR risk assessment model[54] was proposed by Arghandeh *et al.*[25]. They claimed that, contrary to a risk assessment framework, the temporal dimension of disturbances and response time of remedies are to be considered in a resilience framework. Moreover, probabilities of occurrence of disturbance are not crucial except if the system has not yet recovered from a previous disturbance. A resilient system life cycle consists in three steps: (1) system identification, which is the establishment of network topology, physical characteristics, system behaviors, etc. (2) vulnerability analysis, which is basically an ongoing risk analysis taking into consideration the temporal aspect of the disruptions; and (3) resilience operations, which define new settings to improve recovery and absorbing potentials of the system. Once these changes have been made, a new identification phase begins.

Linkov *et al.*[55,56] provided a 4×4 matrix of resilience metrics. Each cell of the matrix corresponds to one of the four stages of event management cycle and one of the four system domains. Domains are different system layers: physical, information, cognitive, and social, and the stages correspond to one pre-event phase (Prepare) and three event handling phases (Absorb, Recover, and Adapt). Instead of providing a metric for resilience, the authors proposed to use cells of the matrix as guidelines to build metrics that, once combined, allow measuring the overall system resilience.

Table 2. Table of resilience evaluations.

Reference	Metrics					Frameworks
	Event specific	Quantitative probabilistic	Quantitative deterministic	Fuzzy	Adversary	
Abimbola and Khan [46]	✓	✓				
Thompson *et al.* [45]	✓	✓			✓	
Babiceanu and Seker [41]	✓	✓	✓			
Francis and Bekera [15]	✓	✓	✓			
Ayyub [27]	✓		✓			
Cai *et al.* [38]	✓		✓			
Gholami *et al.* [40]	✓		✓			
Rieger [39]	✓		✓			
Vugrin *et al.* [26]	✓		✓			
Wei and Ji [34]	✓		✓			
Clark and Sonouz [24]	✓		✓		✓	
Sterbenz *et al.* [42]	✓		✓		✓	✓
Holling [11]		✓				
Shirali *et al.* [43]			✓			
Azadeh *et al.* [50]				✓		
Aleksic *et al.* [49]				✓	✓	
Clédel *et al.* [51]				✓	✓	
Muller [48]				✓	✓	
Linkov *et al.* [55,56]						✓
Sterbenz *et al.* [21]					✓	✓
Mauthe *et al.* [2]					✓	
Van Mieghem *et al.* [64]						✓

The *ResiliNets* strategy [21] is an architectural framework intended to enhance resilience of networks. This framework is based on four axioms: (1) faults are inevitable; (2) normal operation has to be understood; (3) adverse events have to be expected and prepared for; and (4) responses to adverse events are required. According to these axioms, the *ResiliNets* strategy consists in two active phases. The first phase is composed of four steps that are defending, detecting, remediating, and recovering from challenges and attacks, while the second phase enables long-term evolution of the system through diagnostic of the root cause of the fault/attack and refinement of the system behavior to improve the first phase mechanisms and thus to increase the system resilience.

4.5. Adversarial events

Most contemporary control systems have been designed according to conventional model paradigms that are system safety and risk assessment. Originally, these approaches only consider unexpected but accidental events such as human errors or natural disasters. However, the emergence of cyber-physical systems and the accessibility from the Internet of legacy equipment, reliable but not secured, imply that faults resulting from the cyber-environment must be considered. However, only a few approaches presented in this article are able to take these threats into consideration. Indeed, adversarial impacts are explicitly represented in the linear time-invariant model that corresponds to a system [24]. According to Thompson *et al.* [45], resilience only concerns the handling of security breaches. As a consequence, this concept of resilience implies the management of adversarial events. Other approaches (see, e.g., [42,48,49,51]) do not represent events that could impact a system but focus on system's capacities and potentials that are available to handle events. This way, the specific case of adversarial events can be considered without having to explicitly represent them. The counterpart is the inefficiency of such approaches to assess the resilience of a system for a given perturbation. A classification of resilience evaluations is provided in Table 2.

5. RESILIENCE COMPARED WITH OTHER NOTIONS

The term "resilience" is frequently used as a synonym of fault-tolerance [57], adaptive systems [58,59], self-healing [60,61], *etc.* However, resilience is a design paradigm for large scale and complex systems that encompass cybersecurity, physical security, economic efficiency, and dynamic stability [39]. Wei and Ji [34] con-

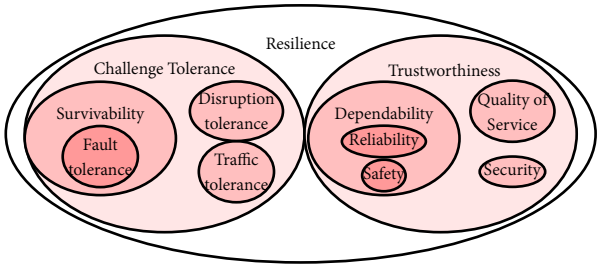


Figure 3. Disciplines of resilience from [21].

sidered resilience as a super-set of numerous properties such as robustness, adaptiveness, survivability, and fault-tolerance. Numerous disciplines contribute to the resilience of a system, but they have been developed independently in different engineering domains [21]. Interconnections between these disciplines are shown in Figure 3 and the Table 2.

5.1. Risk assessment

McDonald [17] described resilience as a capacity to anticipate and manage risk efficiently. However, resilience is clearly distinguished from risk assessment [15,18]. While risk assessment determines potential undesired events, their causal factors and negative consequences, and how to mitigate the exposure of the system to those events, resilience focuses on the system abilities to face undesired events and does not put the emphasis on the events themselves. In the domain of engineered system, safety and resilience are distinct but linked. According to Francis and Bekera [15], resilience aims to compensate poor system design in the case of unanticipated events. As a consequence, resilience can be seen as an addition to safety since it brings the "ability to anticipate, circumvent and recover rapidly from events that threaten safety". Comforting this distinction, the risk assessment goal is situation awareness and diagnostics while "resilience is about the mitigation of unexpected rare extreme failures" [25] that can necessitate extreme remedial actions such as partial or temporary outages in order to ensure the availability of critical services. Resilience is "essential when risk is incomputable" and is characterized "by surprise, complexity, urgency and the necessity of adaptation" [55]. Moreover, historic data of such rare events are out-of-date, uncertain, and biased, and it is not always pertinent to compare them with more recent events [18]. Thus, resilience approaches are complementary to, but distinct from risk analysis approaches, or from risk-aware approaches [62].

On top of that, faults resulting from the cyber-environment and intelligent adversary are generally not considered while critical infrastructure are increasingly connected and cyber-physical systems become the norm [39].

5.2. Robustness

Robustness, as described by Sterbenz *et al.* [21], is a system property that corresponds to the behavior of a system in face of challenges. It bridges the gap between the trustworthiness of a system, which consists in its dependability, security, and quality of service, and the challenge tolerance of the system, which corresponds to the system tolerance to faults, disruptions, intrusion, etc. While resilience and robustness are similar according to Sterbenz *et al.*, other authors make a clear distinction between these two notions.

According to Arghandeh *et al.* [25], "robustness is the ability of a system to cope with a given set of disturbances and maintain its functionality". Thus, robustness is centralized on stability and the handling specific threats, whereas resilience is concerned with flexibility and unbounded perturbations. In other words, resilience tolerates a degradation of performance as it is the ability to recover an original level of performance after a disruption, but, by definition, robustness does not tolerate degradation of performance [56]. The authors of [34,63] compared robustness and resilience: the former is related to consequences and uncertainties given a fixed harmful event while the latter is related to consequences and associated uncertainties but without con-

sidering a specific threat or considering all possible threats. In other words, uncertainties and amplitudes of events are quantified and bounded in robustness discipline and a robust solution can be found according to these quantities. On the other hand, resilience discipline cannot consider these quantities—uncertainties and amplitudes—as harmful events are unknown.

Another definition of robustness is used for networks. The network robustness is defined[64] as: "A measure of the network's response to perturbations or challenges (such as failures or external attacks) imposed on the network". Van Mieghem *et al.* introduced a mathematical value in the interval [0,1], called the R-Value, which is proposed to give a computation of the robustness value of a network.

5.3. Control theory

Several mathematical models, such as differential equations or state-space representation, can be used to model cyber-physical systems[65]. It is well known that, from a differential equation, which models the relation between the inputs and the outputs of a system, we can obtain a state–space representation:

$$x(t+1) = Ax(t) + Bu(t) \tag{7}$$

$$y(t) = Cx(t) + Du(t) \tag{8}$$

In Equation (7), x is a state vector. u and y are, respectively, the input and output vectors. A, B, C, and D are four matrices, respectively, named: state, input, output, and feedthrough matrices. In Equation (8), the output vector y contains the measurements of several sensors. By incorporating and diversifying the sensors to a system, we have more observability. This observability is very useful, especially for the attack detection.

Another important notion is the controllability, which can be defined as follows: our ability to bring a system into a desired state. In fact, incorporating a controller into a cyber-physical system is a way to improve the controllability. The controller uses the outputs of the system to generate the input signal(s). A CPS is a plant which communicates with the physical and the virtual world[66]. To be protected, the design of a CPS aims at controllability and observability. Designing CPS by incorporating physical elements which give controllability and observability can be considered as a way to improve the resilience.

5.4. Other notions

Wei and Ji compared resilience and adaptivity[34]. However, they considered adaptivity limited, as it only concerns mitigation mechanisms that control algorithm parameters, while resilience is open to a larger range of mechanisms. Particularly, adaptivity, as well as fault-tolerance and robustness, does not address the restorability of a system.

Fault-tolerance is the ability of a system to tolerate faults in order to avoid service failures. Sterbenz *et al.*[21] claimed that fault-tolerance is a subset of survivability which considers multiple correlated failures while fault-tolerance does not. It relies on redundancy and is one of the oldest resilience discipline. Moreover, fault-tolerance does not address intelligent adversaries and thus is not sufficient to provide resilience[34].

Morel *et al.*[8] claimed that there is a link between safety and performance levels: any increase in safety is to the detriment of performance. However, resilience lies in this link, and, by tolerating a variation across time of the expected performance level, it is possible to increase the safety level when needed. Resilience is depicted as the gain of safety when performance level is opened to variation.

De Florio[67] considered resilience as "a system's ability to retain certain characteristics of interest", in order to maintain the system identity. This article also introduces elasticity, a complementary notion to resilience, which

considers the system's abilities to change "with respect to its surroundings". Thus, by taking into account these two notions, a new notion, called anti-fragility, can be developed. Anti-fragility encompasses both resilience and elasticity.

6. CONCLUSION

6.1. Gaps and limitations

Most definitions and metrics described in this paper have one thing in common: they derive from risk analysis. According to risk analysis, possible threats can be identified, evaluated, and, even if they are uncertain, their probabilities of occurrence can be estimated. Thereby, resilience is calculated from the results of this risk analysis. Nonetheless, if one tries to assess the resilience of critical infrastructures nowadays, cyber-physical systems and their specific vulnerabilities must be considered. Adversary models must be studied as threats are not only accidental but also come from cyber-criminals, disgruntled employees, and terrorism[68]. These threats from malicious origin are difficult to evaluate. Their probabilities of occurrence are unknown because of the varied nature of the attackers and because of a lack of historical data. Besides, their consequences on the targeted system are hardly predictable.

In addition, several definitions and metrics delegate the evaluation of resilience to an evaluation of service delivery or to an evaluation of system performance. Some articles describe resilience in domain specific terms and provide accurate metrics that match the chosen definition. For example, network resilience is not only concerned with network connectivity[59,69] but also focuses on latency and route stability[58]. However, more generic approaches do not always clearly describe what are system services and system performance. Only a few models (see, e.g.,[51]) provide a framework that makes the description of system services possible.

Another noteworthy remark is the usefulness of the binary assessment of the resilience of a system. It is still critical to predict the behavior of a system when it is challenged by a determined event. This assessment makes it possible to determine if the system is resilient to this event. However, this kind of approach could be less pertinent if the threat is not well defined: its probability of occurrence is vague, its detection is uncertain, and its dynamic behavior, as well as the system response to this threat, are unclear. The authors of[48–51] suggested that assessing the resilience potential of a system could be more relevant than determining whether a system is resilient. Fuzzy logic is used by all four groups to describe this potential for resilience, but other approaches may be considered to assess resilience in a non-binary way.

6.2. Concluding remarks

Many definitions and metrics of resilience are addressed in this paper, from the original definition given by Holling about the resilience in ecological system to more recent and less domain specific ones. Definitions are classified according to their focus: Is resilience defined as the expected behavior when facing attacks and failures or as the combination of systems capacities that allow the mitigation of unexpected events? In addition to the intrinsic system characteristics, is resilience also specific to a determined perturbation? Some of these questions can be used again to classify metrics for resilience. Some metrics are event specific, which implies that resilience of a system must be evaluated separately for every threat or that resilience of a system is the sum of its resilience values for determined threats. Others do not consider possible events and evaluate resilience only from internal characteristics and properties of a system. While the results produced by some metrics determine a timely dependent likelihood of a system to be resilient, others give a resilient score or provide guidelines that ensure the maintenance and the enhancement of system resilience.

To conclude, resilience is compared to some other concepts or paradigms, such as robustness and risk assessment. While it is agreed that resilience is distinct from risk assessment and can be implemented and studied as a complement for traditional design and management approaches, the distinction with other notions is

Table 3. Glossary : Resilience definitions

Notion	Ref.	Title
Origins	[13]	The children of Kauai A longitudinal study from the prenatal period to age ten
	[11]	Resilience and Stability of Ecological Systems
	[14]	Resilience and Vulnerability Adaptation in the Context of Childhood Adversities
	[12]	Resilience: The emergence of a perspective for social-ecological systems analyses
A system property	[11]	Resilience and Stability of Ecological Systems
	[19]	Designing resilient engineered systems
	[16]	Resilient control systems: Next generation design research
	[15]	A metric and frameworks for resilience analysis of engineered and infrastructure systems
	[20]	Resilient control for critical infrastructures and systems
	[17]	Organisational resilience and industrial risk
	[18]	Properties of resilient organizations: an initial view
Service delivery	[23]	Basic concepts and taxonomy of dependable and secure computing
	[22]	From dependability to resilience
	[21]	Resilience and survivability in communication networks: Strategies, principles, and survey of disciplines
	[25]	On the definition of cyber-physical resilience in power systems
	[24]	Cyber-Physical Resilience: Definition and Assessment Metric
Events handling	[28]	On the Definition of Resilience in Systems
	[26]	A resilience assessment framework for infrastructure and economic systems: Quantitative and qualitative resilience analysis of petrochemical supply chains to a hurricane
	[29]	Resilience engineering of industrial processes: Principles and contributing factors
	[27]	Systems Resilience for Multihazard Environments: Definition, Metrics, and Valuation for Decision Making
	[2]	Disaster-Resilient Communication Networks: Principles and Best Practices
Other definitions	[33]	A proposed resilience framework

Table 4. Glossary : Resilience properties

Notion	Ref.	Title
Absorbability	[36]	Resilience for the Scalability of Dependability
	[22]	From dependability to resilience
	[26]	A resilience assessment framework for infrastructure and economic systems: Quantitative and qualitative resilience analysis of petrochemical supply chains to a hurricane
	[29]	Resilience engineering of industrial processes: Principles and contributing factors
	[20]	Resilient control for critical infrastructures and systems
	[25]	On the definition of cyber-physical resilience in power systems
	[35]	Essential characteristics of resilience
Adaptability	[36]	Resilience for the Scalability of Dependability
	[37]	Validation of innovative state estimation and control techniques
	[22]	From dependability to resilience
	[26]	A resilience assessment framework for infrastructure and economic systems: Quantitative and qualitative resilience analysis of petrochemical supply chains to a hurricane
	[20]	Resilient control for critical infrastructures and systems
	[30]	Defining resilience
	[35]	Essential characteristics of resilience
Recoverability	[26]	A resilience assessment framework for infrastructure and economic systems: Quantitative and qualitative resilience analysis of petrochemical supply chains to a hurricane
Other capacities, descriptions	[36]	Resilience for the Scalability of Dependability
	[22]	From dependability to resilience
	[34]	Resilient industrial control system (RICS): Concepts, formulation, metrics, and insights
	[21]	Resilience and survivability in communication networks: Strategies, principles, and survey of disciplines
	[29]	Resilience engineering of industrial processes: Principles and contributing factors
	[33]	A proposed resilience framework

not always trivial. For example, even if some authors do not differentiate robustness and resilience in theory, the fact that these notions had originally been developed in independent scientific domains and in different communities produces a difference of usage in practice.

Table 5. Glossary : Metrics for resilience

Notion	Ref.	Title
Quantitative deter.	[11]	Resilience and Stability of Ecological Systems
	[4]	Micro incident analysis framework to assess safety and resilience in the operation of safe critical systems: A case study in a nuclear power plant
	[34]	Resilient industrial control system (RICS): Concepts, formulation, metrics, and insights
	[26]	A resilience assessment framework for infrastructure and economic systems: Quantitative and qualitative resilience analysis of petrochemical supply chains to a hurricane
	[42]	Modelling and analysis of network resilience
	[15]	A metric and frameworks for resilience analysis of engineered and infrastructure systems
	[39]	Resilient control systems Practical metrics basis for defining mission impact
	[27]	Systems Resilience for Multihazard Environments: Definition, Metrics, and Valuation for Decision Making
	[38]	Availability-based engineering resilience metric and its corresponding evaluation methodology
	[40]	Toward a Consensus on the Definition and Taxonomy of Power System Resilience
	[24]	Cyber-Physical Resilience: Definition and Assessment Metric
	[41]	Cyber resilience protection for industrial internet of things: A software-defined networking approach
Semi-quantitative		A new method for quantitative assessment of resilience engineering by PCA and NT approach:
	[43]	A case study in a process industry
	[18]	Properties of resilient organizations: an initial view
Quantative prob.	[11]	Resilience and Stability of Ecological Systems
Event specific	[44]	On the Complex Definition of Risk: A Systems-Based Approach
	[28]	On the Definition of Resilience in Systems
	[34]	Resilient industrial control system (RICS): Concepts, formulation, metrics, and insights
	[42]	Modelling and Analysis of Network Resilience
	[15]	A metric and frameworks for resilience analysis of engineered and infrastructure systems
	[33]	A proposed resilience framework
	[45]	A New Resilience Taxonomy
	[41]	Cyber resilience protection for industrial internet of things: A software-defined networking approach
	[46]	Resilience modeling of engineering systems using dynamic objectoriented Bayesian network approach
Fuzzy models	[47]	Fuzzy sets
	[52]	A Comparison of Commercial and Military Computer Security Policies
	[53]	Automated support for external consistency
	[48]	Fuzzy Architecture Assessment for Critical Infrastructure Resilience
	[49]	An assessment of organizational resilience potential in SMEs of the process industry, a fuzzy approach
		A new method for quantitative assessment of resilience engineering by PCA and NT approach
	[43]	A case study in a process industry
	[15]	A metric and frameworks for resilience analysis of engineered and infrastructure systems
	[50]	Assessment of resilience engineering factors in high-risk environments by fuzzy cognitive maps A petrochemical plant
	[18]	Properties of resilient organizations: an initial view
	[51]	Towards the Evaluation of End-to-End Resilience Through External Consistency
Frameworks	[54]	At Risk: Natural Hazards, People's Vulnerability and Disasters
	[21]	Resilience and survivability in communication networks: Strategies, principles, and survey of disciplines
	[64]	A Framework for Computing Topological Network Robustness
	[42]	Modelling and Analysis of Network Resilience
	[55]	Measurable Resilience for Actionable Policy
	[56]	Resilience metrics for cyber systems
	[25]	On the definition of cyber-physical resilience in power systems
	[2]	Disaster-Resilient Communication Networks: Principles and Best Practices
Adversarial events		Resilience and survivability in communication networks:
	[21]	Strategies, principles, and survey of disciplines
	[42]	Modelling and analysis of network resilience
	[48]	Fuzzy Architecture Assessment for Critical Infrastructure Resilience
	[49]	An assessment of organizational resilience potential in SMEs of the process industry, a fuzzy approach
	[45]	A New Resilience Taxonomy
	[2]	Disaster-Resilient Communication Networks: Principles and Best Practices
	[51]	Towards the Evaluation of End-to-End Resilience Through External Consistency
	[24]	Cyber-Physical Resilience: Definition and Assessment Metric

Designing resilient systems is a challenge, especially in the case of CPS used in critical infrastructures. As described in Section 5, intrinsic properties of a CPS can be used to include, for example, physical components, making the system resilient by design. These components can be considered as protective layers for the CPS. One of the actual challenges consists in improving a CPS resilience by diversifying its incorporated hardware, or software components.

To provide an overall view of the main notions included in this paper, we refer the reader to the three glossaries, respectively, related to: resilience definitions [Table 3], resilience properties [Table 4], and resilience metrics [Table 5]. Based on the observations made, and on the classifications of the existing definitions, properties, and metrics, there are several topics that can be addressed in future works.

DECLARATIONS

Authors' contributions

Wrote and review the article: Clédel T, Cuppens N, Cuppens F,Dagnas R.
Each author contributed equally to the paper.

Availability of data and materials

Not applicable.

Financial support and sponsorship

This work was supported by the Cyber CNI Chair of Institute Mines-Télécom which is held by IMT Atlantique and supported by Airbus Defence and Space, Amossys, BNP Parisbas, EDF, Nokia and the Regional Council of Brittany; it has been acknowledged by the French Centre of Excellence in Cybersecurity.

Conflicts of interest

All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

REFERENCES

1. Peng Y, Lu T, Liu J, Gao Y, Guo X, Xie F. Cyber-physical System Risk Assessment, 2013 Ninth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2013,Oct. 16-18, Beijing, China. IEEE, 2013. pp. 442-7.
2. Mauthe A, Hutchison D, Çetinkaya EK, et al. Disaster-resilient communication networks: Principles and best practices, 2016 8th International Workshop on Resilient Networks Design and Modeling (RNDM), 2013 Oct. 16-18, Halmstad, Sweden. IEEE, 2016. pp. 1-10.
3. Hollnagel E. Resilience: The challenge of the unstable. In: Hollnagel E, Woods D, Leveson N, editors. Resilience engineering: Concepts and precepts, 1st ed. Aldershot: Ashgate; 2006. pp. 9-17.
4. Carvalho P V R, dos Santos I L, Gomes J O, Borges M R S. Micro incident analysis framework to assess safety and resilience in the operation of safe critical systems: A case study in a nuclear power plant. *J Loss Prevent Proc* 2008;21:277–86.
5. Saurin T A, Carim Júnior G C. Evaluation and improvement of a method for assessing HSMS from the resilience engineering perspective: A case study of an electricity distributor. *Saf Sci* 2011;49:355–68.
6. Azadeh A, Salehi V, Ashjari B, Saberi M. Performance evaluation of integrated resilience engineering factors by data envelopment analysis: The case of a petrochemical plant. *Proc Saf Environ Protec* 2014;92:231–41.
7. Shirali G H A, Motamedzade M, Mohammadfam I, Ebrahimipour V, Moghimbeigi A. Challenges in building resilience engineering (RE) and adaptive capacity: A field study in a chemical plant. *Process Saf Environ* 2012;90:83–90.
8. Morel G, Amalberti R, Chauvin C. How good micro/macro ergonomics may improve resilience, but not necessarily safety. *Saf Sci* 2009;47:285–94.
9. Abech M P, Berg G A, Delis M G, Guimaraes L B M, Woods D D, editors. Analyzing Resilience of an Oil Distribution Plant. Proceedings of the 2006 IEEE Systems and Information Engineering Design Symposium; 2006 April 28-28; Charlottesville, VA, USA. IEEE; 2007.
10. Hale A, Heijer T. Is resilience really necessary? The case of railways. In: Hollnagel E, Woods D, Leveson N, editors. Resilience engineering: Concepts and precepts, 1st ed. Aldershot: Ashgate; 2006. pp. 125-48
11. Holling C S. Resilience and Stability of Ecological Systems. *Annu Rev Ecol Syst* 1973;4:1–23.

12. Folke C. Resilience: The emergence of a perspective for social–ecological systems analyses. *Glob Environ Change* 2006;16:253–67.
13. Werner E E, Bierman J M, French F E. The children of Kauai: A longitudinal study from the prenatal period to age ten. Honolulu: University of Hawaii Press; 1971.
14. Luthar S S, editor. Resilience and Vulnerability: Adaptation in the Context of Childhood Adversities. Cambridge: Cambridge University Press; 2003. [DOI: 10.1017/CBO9780511615788]
15. Francis R, Bekera B. A metric and frameworks for resilience analysis of engineered and infrastructure systems. *Reliab Eng Syst Safe* 2014;121:90–103.
16. Rieger C G, Gertman D I, McQueen M A, editors. Resilient control systems: Next generation design research. Proceedings of the 2009 2nd Conference on Human System Interactions; 2009 May 21-23; Catania, Italy. IEEE; 2009.
17. McDonald N. Organisational resilience and industrial risk. In: Hollnagel E, Woods D, Leveson N, editors. Resilience engineering: Concepts and precepts, 1st ed. Aldershot: Ashgate; 2006. pp. 155–80.
18. Wreathall J. Properties of resilient organizations: an initial view. In: Hollnagel E, Woods D, Leveson N, editors. Resilience engineering: Concepts and precepts, 1st ed. Aldershot: Ashgate; 2006. pp. 275–85.
19. Mitchell S M, Mannan M S, O'Connor M K. Designing resilient engineered systems. *Chem Eng Prog* 2006;102:39–15.
20. Yang Y, Syndor R. Resilient control for critical infrastructures and systems. NRC 2014. Available from: https://www.researchgate.net/profile/Yaguang_Yang/publication/283091635_Resilient_control_for_critical_infrastructures_and_systems/links/562a9ec108ae518e347f74e1/Resilient-control-for-critical-infrastructures-and-systems. [Last accessed on 04-10-2020]
21. Sterbenz J P G, Hutchison D, Çetinkaya E K, et al. Resilience and survivability in communication networks: Strategies, principles, and survey of disciplines. *Com Net* 2010;54:1245–65.
22. Laprie J C. From dependability to resilience. Available from: https://www.ece.cmu.edu/~koopman/dsn08/fastabs/dsn08fastabs_laprie.pdf.. [Last accessed on 04-10-2020]
23. Avizienis A, Laprie J C, Randell B, Landwehr C. Basic concepts and taxonomy of dependable and secure computing. *IEEE T Depend Secure* 2004;1:11-33.
24. Clark A, Zonouz S. Cyber-Physical Resilience: Definition and Assessment Metric. *IEEE T Smart Grid* 2019;10:1671-84.
25. Arghandeh R, von Meier A, Mehrmanesh L, Mili L. On the definition of cyber-physical resilience in power systems. *Renew Sust Energ Rev* 2016;58:1060–9.
26. Vugrin E D, Warren D E, Ehlen M A. A resilience assessment framework for infrastructure and economic systems: Quantitative and qualitative resilience analysis of petrochemical supply chains to a hurricane. *Proc Safety Prog* 2011;30:280-90.
27. Ayyub B M. Systems Resilience for Multihazard Environments: Definition, Metrics, and Valuation for Decision Making. *Risk Anal* 2014;34:340-55.
28. Haimes Y Y. On the Definition of Resilience in Systems. *Risk Anal* 2009;29:498-501.
29. Dinh L T, Pasman H, Gao X, Mannan M S. Resilience engineering of industrial processes: Principles and contributing factors. *J Loss Prevent Proce Indus* 2012;25:233-41.
30. Hale A, Heijer T. Defining resilience. In: Hollnagel E,Woods D, LevesonN, editors. Resilience engineering: Concepts and precepts, 1st ed. Aldershot: Ashgate; 2006. pp. 35–40.
31. Leveson N, Dulac N, Zipkin D, Cutcher-Gershenfeld J, Carroll J, Barrett B. Engineering resilience into safety-critical systems. In: Hollnagel E,Woods D, Leveson N, editors. Resilience engineering: Concepts and precepts, 1st ed. Aldershot: Ashgate; 2006. pp. 95–123.
32. Sundström G, Hollnagel E. Learning how to create resilience in business systems. In: Hollnagel E, Woods D, Leveson N, editors. Resilience engineering: Concepts and precepts, 1st ed. Aldershot: Ashgate; 2006. pp. 235–52.
33. Thompson M A, Ryan M J, McLucas A C, editors. A proposed resilience framework. Proceedings of the Systems Engineering and Test and Evaluation (SETE) Conference; 2014 April; Canberra, AS. 2002. Available from: https://www.researchgate.net/profile/Mike_Ryan7/publication/274660820_A_Proposed_Resilience_Framework/links/5524ff4c0cf22e181e73b971.pdf. [Last accessed on 04-10-2020]
34. Wei D, Ji K, editors. Resilient industrial control system (RICS): Concepts, formulation, metrics, and insights. Proceedings of the 2010 3rd International Symposium on Resilient Control Systems; 2010 Aug 10-12; Idaho Falls, ID, USA. IEEE; 2010.
35. Woods D D. Essential characteristics of resilience. In: Hollnagel E, Woods D, Leveson N, editors. Resilience engineering: Concepts and precepts, 1st ed. Aldershot: Ashgate; 2006. pp. 21–34.
36. Laprie J-C, editor. Resilience for the Scalability of Dependability. Proceedings of the 4th IEEE International Symposium on Network Computing and Applications; 2005 July 27-29; Cambridge, MA, USA. IEEE; 2006.
37. de Lafontaine J, Côté J, Kron A, Vuilleumier P, Santandrea S, van den Braembussche P, editors. Validation of innovative state estimation and control techniques on PROBA-2. Proceedings of the 6th International ESA Conference on Guidance, Navigation and Control Systems. 2005 Oct 17-20; Loutraki, Greece. ESA SP-606; 2006. Available from: http://adsabs.harvard.edu/full/2006ESASP.606E..23D. [Last accessed on 04-10-2020]
38. Cai B, Xie M, Liu Y, Liu Y, Feng Q. Availability-based engineering resilience metric and its corresponding evaluation methodology. *Reliability Engineering & System Safety* 2018;172:216-24.
39. Rieger C G, editor. Resilient control systems Practical metrics basis for defining mission impact. Proceedings of the 2014 7th International Symposium onResilient Control Systems (ISRCS); 2014Aug 19-21; Denver, CO, USA. IEEE; 2014.
40. Gholami A, Shekari T, Amirioun MH, Aminifar F, Amini MH, Sargolzaei A. Toward a Consensus on the Definition and Taxonomy of Power System Resilience. *IEEE Access* 2018;6:32035-53.
41. Babiceanu R F, Seker R. Cyber resilience protection for industrial internet of things: A software-defined networking approach. *Computers in Industry* 2019;104:47-58.

42. Sterbenz J P G, Çetinkaya E K, Hameed M A, Jabbar A, Rohrer J P, editors. Modelling and analysis of network resilience. Proceedings of the 2011 Third International Conference on Communication Systems and Networks (COMSNETS 2011); 2011 Jan 4-8; Bangalore, India. IEEE; 2011.
43. Shirali G A, Mohammadfam I, Ebrahimipour V. A new method for quantitative assessment of resilience engineering by PCA and NT approach: A case study in a process industry. *Reliability Engineering & System Safety* 2013;119:88-94.
44. Haimes Y Y. On the Complex Definition of Risk: A Systems-Based Approach. *Risk Anal* 2009;29:1647-54.
45. Thompson M A, Ryan M J, Slay J, McLucas A C. A New Resilience Taxonomy. *INCOSE International Symposium* 2016;26:1318-30.
46. Abimbola M, Khan F. Resilience modeling of engineering systems using dynamic object-oriented Bayesian network approach. *Computers & Industrial Engineering* 2019;130:108-18.
47. Zadeh L A. Fuzzy Sets. *Information and Control* 1965;8:338-53.
48. Muller G. Fuzzy Architecture Assessment for Critical Infrastructure Resilience. *Procedia Computer Science* 2012;12:367-72.
49. Aleksić Al, Stefanović M, Arsovski S, Tadić D. An assessment of organizational resilience potential in SMEs of the process industry, a fuzzy approach. *J Loss Prevent Proce Industr* 2013;26:1238-45.
50. Azadeh A, Salehi V, Arvan M, Dolatkhah M. Assessment of resilience engineering factors in high-risk environments by fuzzy cognitive maps: A petrochemical plant. *Saf Sci* 2014;68:99-107.
51. Clédel T, Foley S N, Cuppens N, Cuppens F, Kermarrec Y, et al. Towards the Evaluation of End-to-End Resilience Through External Consistency. In: Castiglione A, Pop F, Ficco M, Palmieri F, editors. Proceedings of the 10th International Symposium on Cyberspace and Security (CSS); 2018 Oct 29-31; Amalfi, Italy. Springer; 2018. pp. 99-114.
52. Clark D D, Wilson D R. A Comparison of Commercial and Military Computer Security Policies. Proceedings of the 1987 IEEE Symposium on Security and Privacy; 1987 April 27-29; Oakland, CA, USA. IEEE; 2014.
53. Williams J G, Padula L J L. Automated support for external consistency. Proceedings of the [1993] Proceedings Computer Security Foundations Workshop VI; 1993 June 15-17; Franconia, NH, USA. IEEE; 2002.
54. Wisner B, Blaikie P M, Cannon T, Davis I. At Risk: Natural Hazards, People's Vulnerability and Disasters. 2nd ed. London: Routledge; 2004. Available from: https://books.google.fr/books?id=566bdm7T5VEC. [Last accessed on 04-10-2020]
55. Linkov I, Eisenberg DA, Bates ME, et al. Measurable Resilience for Actionable Policy. *Environ Sci Technol* 2013;47:10108-10110.
56. Linkov I, Eisenberg DA, Plourde K, Seager TP, Allen J, Kott A. Resilience metrics for cyber systems. *Environ Syst Decis* 2013;33:471-6.
57. Stoller S D, Liu Y A. Algorithm Diversity for Resilient Systems. In: Foley SN, editor. Proceedings of the IFIP Annual Conference on Data and Applications Security and Privacy; 2019 Jul 15-17; Charleston, SC, USA. Springer; 2019. pp. 359-378.
58. Andersen D, Balakrishnan H, Kaashoek F, Morris R. Resilient Overlay Networks. *Sigcomm Comput Commun Rev* 2002;32:66-66.
59. Costa da Fontoura L. Reinforcing the resilience of complex networks. *Phys Rev E* 2004;69:066127.
60. Sousa P, Neves N F, Verissimo P, editors. How resilient are distributed f fault/intrusion-tolerant systems?. Proceedings of the 2005 International Conference on Dependable Systems and Networks (DSN'05); 2005 June 28 - Jul 1; Yokohama, Japan. IEEE; 2005.
61. Lucia W, Sinopoli B, Franze G, editors. A set-theoretic approach for secure and resilient control of Cyber-Physical Systems subject to false data injection attacks. Proceedings of the 2016 Science of Security for Cyber-Physical Systems Workshop (SOSCYPS); 2016 April 11-11; Vienna, Austria. IEEE; 2016.
62. Kanoun W, Cuppens-Boulahia N, Cuppens F, Dubus S, editors. Risk-aware Framework for Activating and Deactivating Policy-based Response Risk-aware framework for activating and deactivating policy-based response. Proceedings of the 2010 Fourth International Conference on Network and System Security; 2010 Sept 1-3; Melbourne, VIC, Australia. IEEE; 2010.
63. Aven T. On Some Recent Definitions and Analysis Frameworks for Risk, Vulnerability, and Resilience. *Risk Anal* 2011;31:515-522.
64. Van Mieghem P, Doerr C, Wang H, Hernandez J M, Hutchison D, et al. A framework for computing topological network robustness. Delft University of Technology, Report 20101218, 2010.
65. Baheti R, Gill H. Cyber-physical systems. *The Impact of Control Technology* 2011;12:161-166.
66. Lee E A, editor. Cyber-physical systems-are computing foundations adequate. Proceedings of the Position paper for NSF Workshop on Cyber-physical Systems: Research Motivation, Techniques and Roadmap; 2006 Oct 16-17; Austin, TX, USA.
67. De Florio V. Antifragility = Elasticity + Resilience + Machine Learning: Models and Algorithms for Open System Fidelity. *Procedia Comput Sci* 2014;32:834-41.
68. Cardenas AA, Amin S, Sinopoli B, et al. Challenges for Securing Cyber Physical Systems. 1st Workshop CyberPhys. *Syst Security* 2009;5:1.
69. Hayel Y, Quanyan Z, editors. Resilient and secure network design for cyber attack-induced cascading link failures in critical infrastructures. Proceedings of the 2015 49th Annual Conference on Information Sciences and Systems (CISS); 2015 Mar 18-20; Baltimore, MD, USA. IEEE; 2015.

Corizzo *et al. J Surveill Secur Saf* 2020;1:140-50
DOI: 10.20517/jsss.2020.15

Journal of Surveillance, Security and Safety

Original Article

Open Access

Check for updates

Feature extraction based on word embedding models for intrusion detection in network traffic

Roberto Corizzo[1], Eftim Zdravevski[2], Myles Russell[1], Andrew Vagliano[3], Nathalie Japkowicz[1]

[1]Department of Computer Science, American University, Washington, DC 20016, USA.
[2]Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University, Skopje 1000, North Macedonia.
[3]Department of Computer Science, Northwestern University, Evanston, IL 60208, USA.

Correspondence to: Dr. Roberto Corizzo, Department of Computer Science, American University, 4400 Massachusetts Avenue NW, Washington, DC 20016, USA. E-mail: rcorizzo@american.edu

How to cite this article: Corizzo R, Zdravevski E, Russell M, Vagliano A, Japkowicz N. Feature extraction based on word embedding models for intrusion detection in network traffic. *J Surveill Secur Saf* 2020;1:140-50. http://dx.doi.org/10.20517/jsss.2020.15

Received: 30 Apr 2020 **First Decision**: 15 Jun 2020 **Revised**: 27 Jun 2020 **Accepted**: 17 Jul 2020 **Available online**: 28 Dec 2020

Academic Editor: Xiaofeng Chen **Copy Editor**: Cai-Hong Wang **Production Editor**: Jing Yu

Abstract

Aim: The analysis of network traffic plays a crucial role in modern organizations since it can provide defense mechanisms against cyberattacks. In this context, machine learning algorithms can be fruitfully adopted to identify malicious patterns in network sessions. However, they cannot be directly applied to a raw data representation of network traffic. An active thread of research focuses on the design and implementation of feature extraction techniques that aim at mapping raw data representations of network traffic sessions to a new representation that can be processed by machine learning algorithms.

Methods: In this paper, we propose a feature extraction approach based on word embedding models. The proposed approach extracts semantic features characterized by contextual information that is hidden in the raw data representation.

Results: Our experiments conducted on three datasets showed that our feature extraction approach based on word embedding models has the potential to increase the classification performance of conventional machine learning algorithms that are applied to intrusion detection, and it is competitive with known feature extraction baselines in the state-of-the-art.

Conclusion: This study shows that word embedding models can be used to carry out intrusion detection tasks accurately. Feature extraction based on word embedding models requires a higher computational time than simpler techniques, but leads to a higher accuracy, which is important for the identification of complex attacks.

Keywords: Feature extraction, intrusion detection, network traffic, anomaly detection, word embeddings, language models

INTRODUCTION

Intrusion detection systems (IDS) play a fundamental role in modern organizations, providing defense mechanisms against cyberattacks. IDS monitor and analyze the traffic using different sources of information, with the purpose of identifying intrusions and other security breaches. Differently than firewalls, which limit access between networks to prevent intrusions, IDS evaluate a potential intrusion when it takes place, signal an alarm, and may terminate the connection. The most popular categories of IDS include network-based IDS and host-based IDS (HIDS)[1]. The former analyze network packets on an entire subnet[2,3], whereas the latter consist of an agent on a host that analyzes system calls, file system changes, and logs[4-7]. In this study, we focused on HIDS and, more specifically, machine learning-based tools to support it. One opportunity in this domain consists in monitoring and analyzing network traffic represented in the form of network sessions, also known as traces[8]. One of the most popular data representations for traces is that known as sequence of system calls[9], i.e., a sequence of requests that programs submit to the operating system kernel to perform any action. The ordering, type, length and other attributes of system calls made by an application process can provide a unique signature or trace. Such information is highly informative, and it is exploited in current IDS to help distinguish between normal and abnormal behaviors in a network session[10].

Relevant benchmark datasets such as the Defense Advanced Research Projects Agency dataset[11] and the Knowledge Discovery and Data Mining Tools Competition (KDD'99) dataset[12] have been analyzed in a large number of studies for the past two decades[2-4,13-16]. However, such datasets do not cover up-to-date attack scenarios, and therefore, they are not considered to be challenging at present. More recently, the Australian Defence Force Academy Linux Dataset (ADFA-LD)[5,17,18], as well as the Next-Generation Intrusion Detection System Dataset (NGIDS-DS)[18,19] and the Web Conference 2019 (WWW2019)[20] datasets, succeeded in filling this gap, presenting new and relevant types of attacks conceived to assess the accuracy of modern intrusion detection tools. The datasets present thousands of system call traces collected from a Linux local server, with normal and attack behaviors.

Traditional machine learning algorithms can be fruitfully exploited to identify malicious patterns in network sessions, which can be subsequently filtered. Examples of approaches in the literature include Support Vector Machines[13], Artificial Neural Networks[2], classification of association rules[14,15], decision trees[4], random forests[3], and ensembles of classifiers[16].

However, machine learning algorithms cannot be directly applied to a raw data representation of network traffic, such as sequences of system calls. For this reason, an active thread of recent research[5-7] focuses on the design and implementation of feature extraction techniques that aim at mapping sequences of system calls to a new representation that can be processed by machine learning algorithms. Figure 1 shows the typical analytical workflow that is carried out to perform machine learning-based intrusion detection.

Focusing on feature extraction approaches in the literature, pattern-based and frequency-based methods represent the most popular classes. Pattern-based approaches identify patterns in sessions, consisting of multiple co-occurring system calls in a trace, whereas frequency-based approaches[5,21,22] extract feature vectors in which entries represent the frequency of a system call in a trace. Although the former generally lead to a more accurate profile of the normal class, they are computationally more expensive. On the other hand, the latter are more computationally efficient, but the resulting representation does not take into account the position of system calls in the trace[6].

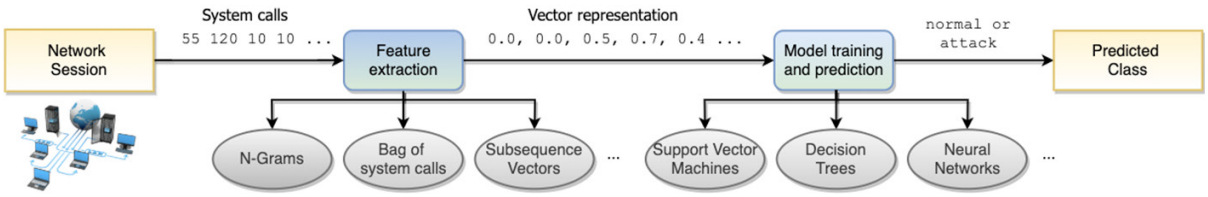


Figure 1. Analytical workflow for machine learning-based intrusion detection in network traffic. Network sessions in the form of sequences of system calls are fed to a feature extraction method, which returns vector data that can be exploited in the modeling step by machine learning and deep learning algorithms. The outcome is a returned class for each session (normal, attack)

One example of a pattern-based approach is the *N-gram* feature extraction method[7], which generates pattern data, converting each class into a two-dimensional array (or into a matrix) representation. In this representation, columns are grams, i.e., attributes, and rows are instances, i.e., traces. The entries in the matrix are the number of occurrences of each N-gram in the traces. Considering that the number of grams for any of the classes is very high compared to the number of instances, it is common to aim for a reduction in the number of attributes, taking into account the most frequent grams.

Focusing on frequency-based approaches, the Subsequence Vector method[5] transforms a trace into a vector, where entries are calculated as the product between the system call and its frequency in the trace. The limitations of this approach consist in the generation of sparse vector representations and in the independent treatment of each system call. Another similar method is known as Bag of System Calls[22], which enumerates all system calls and transforms system traces into fixed-length vectors that contain the frequencies of each system call. One alternative to exploit frequency vectors is to apply weighting schemes to the observed frequencies. This type of approach is followed in the study by Xie *et al.*[5], which proposes the application of Term-Frequency and Inverse Document Frequency (TF-IDF) to extract normalized frequency vectors. Another alternative consists in performing dimensionality reduction to obtain a more compact vector representation that does not present sparsity issues. One example of this type of approach can be found in the study by Xie *et al.*[6], which proposes the application of principal component analysis on frequency vectors.

However, one major challenge in feature extraction is to represent the contextual information of system calls in traces effectively. Contextual information in sequential data with a complex structure can be often hidden and difficult to extract[23,24], especially for pattern-based and frequency-based approaches that do not take into account the temporal dynamics of system calls in traces.

In this paper, we propose a new feature extraction method for sequential network traffic data in the form of sequence of system calls. Following the success of state-of-the-art feature extraction methods inspired by Natural Language Processing (NLP), our method leverages a word embedding-based approach to extract contextual information that can be exploited in the subsequent classification step by any machine learning algorithm. To the best of our knowledge, this is the first study that presents feature extraction based on word embedding models and, in particular, presents a combination approach with TF-IDF and Word2Vec models. Moreover, in our study, we also investigated feature extraction based on Doc2Vec. We performed experiments to evaluate the effectiveness of different machine learning classifiers with our extracted features, and compared them with different state-of-the-art feature extraction methods in a number of different scenarios.

METHODS

In this section, we provide a brief overview on word embedding models and some examples of their successful application. Subsequently, we describe our proposed feature extraction method for intrusion detection in network traffic, based on word embedding models.

Word embedding models are commonly adopted techniques for language modeling and feature learning in NLP. These techniques map words and sentences into low dimensional feature vectors that can be exploited by automated analytical tools. Examples of word embedding techniques include neural networks[25], probabilistic models[26], and approaches based on dimensionality reduction applied to a word co-occurrence matrix[27].

Some word embedding techniques aim at extracting a vector representation for a word in terms of co-occurring words, whereas others express a word in terms of vector of linguistic contexts[28]. Recently, particular interest has been devoted to the latter, since they attempt to characterize the semantics of words and sentences, on the basis of the intuition by which a word is characterized according to the company it keeps[29,30].

One example of a groundbreaking technique in this field is represented by Word2Vec[25]. Its ability to represent implicit relationships between words has resulted in substantial machine learning improvements on domains by contextual information. Some examples include the classification of news articles and tweets[31], the analysis of biological data for the prediction of therapeutic peptides[32], the detection of malware activity on Android devices[33], and the recommendation of contents in social networks[34]. Similarly to these studies, the method proposed in this paper leverages Word2Vec as a method to extract word embeddings. However, none of these approaches applies Word2Vec to network traffic sessions in the form of sequences of system calls. Our aim was to propose a pipeline that makes Word2Vec applicable to data in this domain. In addition, we proposed an approach to weight the feature extracted according to its importance.

The common result obtained in[31,33,34] is that performing the learning task on top of the newly extracted data representation obtained by means of word embedding models, leads to an improved accuracy. The motivation is that the newly extracted representation presents useful semantic features that were hidden in the initial raw data representation, thus facilitating machine learning tools to perform classification and improving the machine learning classification task. Following the same intuition, and motivated by the success in different domains, our proposed method leverages a Word2Vec word embedding model to extract contextual information that can be exploited in the subsequent classification step by any machine learning algorithm. In particular, we exploit Word2Vec to obtain a k-dimensional numerical embedding vector that entails the semantic representation of a system call. Given a set of labeled traces T_L, for which the class attribute is known (normal or attack), we train a Word2Vec model to generate semantic vectors for all traces $t \in T_L$. The feature extraction process from network traces exploiting a Word2Vec model is shown in Figure 2. One alternative to Word2Vec is represented by Doc2Vec, which extracts a unique representation for each document.

The novelty in this paper is to exploit Word2Vec in combination with a TF-IDF model[35]. More specifically, a TF-IDF model is trained to subsequently perform a weighted transformation of the semantic representation of a system call extracted by Word2Vec. The rationale for the adoption of such a model is that the representation vector of a trace should be weighted according to the saliency of the system calls it contains. More precisely, system calls that appear in several traces are less indicative of the content of a trace, whereas system calls that appear rarely, should be more discriminative. The TF-IDF weighting allows us to capture these properties and give more weight to system calls that are frequent in a trace but rare in the overall collection of traces.

Each trace $t \in T_L$ is represented as a bag of system calls $b(t) = \{s_1, s_2, \ldots, s_k\}$ of arbitrary length. Next, the Word2Vec model converts a system call $s \in b(t)$ into a semantic vector $w(s)$ that is multiplied by the TF-IDF score $weight(s)$ calculated as follows:

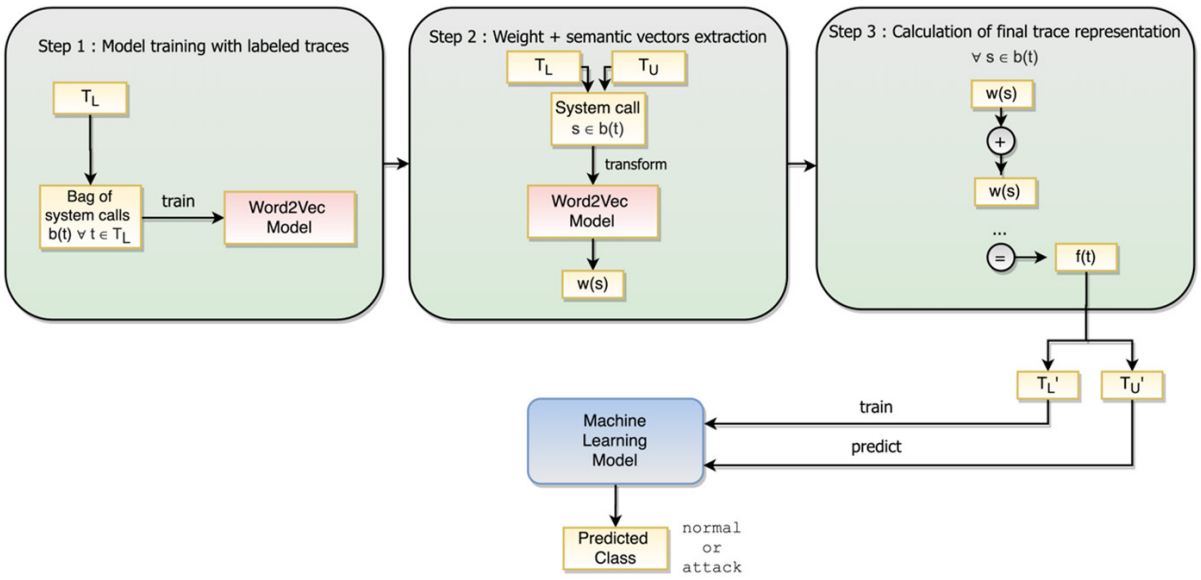


Figure 2. Graphical representation of feature extraction based on a Word2Vec model

$$weight(s) = freq_{s,t} \times log_2 \frac{|T_L|}{freq_{s,T_L}},$$

where $freq_{s,t}$ is the frequency of the system call s in the trace t, and $freq_{s,T_L}$ is the frequency of the system call s in the entire collection. To extract a single vector representation for each trace, we exploit the "additive compositionality" property of word embeddings. This property guarantees that similar words appear close to each other in the feature space, and that the sum of their embedding vector representation resembles an *AND* concatenation. By analogy, in our domain, if two traces (t_1, t_2) appear in the same context, their sum vectors obtained as the sum of the embedding vectors of the corresponding system calls will still be close to each other. Therefore, the final vector representation $f(t)$ of a trace t is computed as:

$$f(t) = \sum_{s \in b(t)} w(s) \times weight(s).$$

Following this process, we obtain a new dataset $T_L' \in \mathbb{R}^{|T_L| \times k}$, consisting of the semantic vector representation $f(t)$ for each labeled trace t in T_L. This dataset can be used to train any machine learning algorithm.

Consequently, during the prediction phase, the previously trained Word2Vec and TF-IDF models are exploited to extract features for a new collection of unlabeled traces T_U. The machine learning algorithm of choice can exploit the extracted representation to predict the class attribute of each trace $u \in T_U$. The overall feature extraction process with Word2Vec and TF-IDF is shown in Figure 3.

In summary, the Word2Vec and TF-IDF models are trained with a collection of labeled traces T_L, represented as a bag of system calls (Step 1). The outputs of these models are combined to extract a new representation (T_L', T_U') from both labeled and unlabeled traces (T_L, T_U). The representation extracted by Word2Vec is a vector for each system call. Simultaneously, the TF-IDF model extracts the weight corresponding to each system call (Step 2). The multiple vectors that represent the different system calls in a trace are subsequently calculated as the weighted sum of the system calls vector representations extracted by Word2Vec and the TF-IDF weights (Step 3). A machine learning model is trained on labeled traces after feature extraction T_L' and predicts the class of unlabeled traces T_U' after the feature extraction process.

In the following section, we present our experiments aimed at comparing the classification accuracy with our proposed feature extraction technique in comparison with state-of-the-art feature extraction methods.

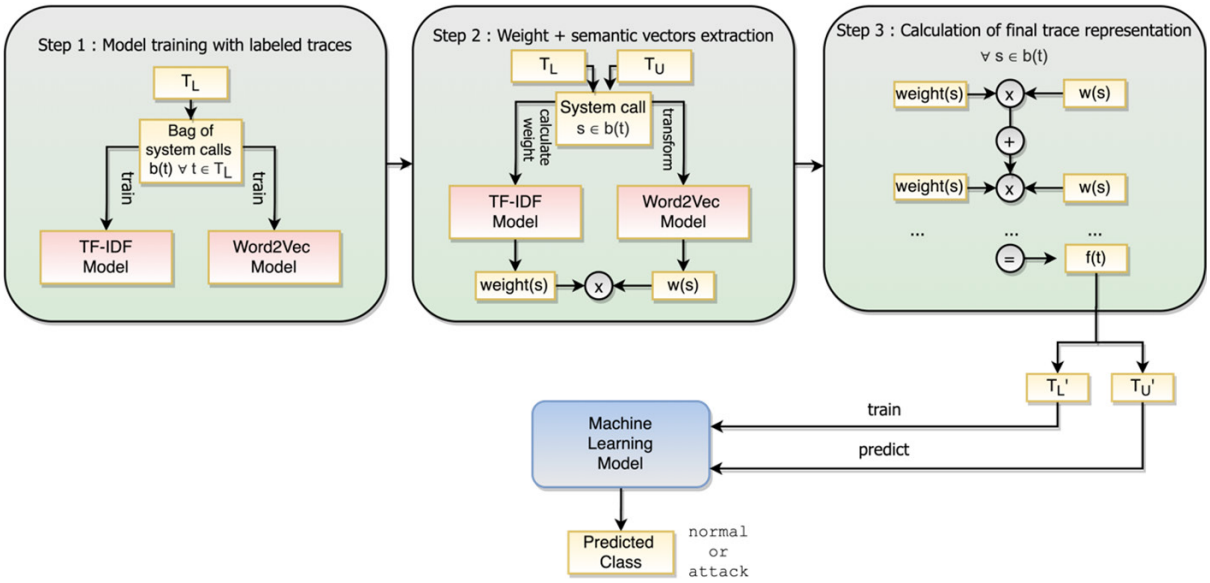


Figure 3. Graphical representation of feature extraction based on Word2Vec and Term-Frequency and Inverse Document Frequency (TF-IDF) models

The implementations of the proposed approach are publicly available in our GitHub repository (https://github.com/rcorizzo/hids-word-embedding). The implementations are available in the Python programming language, and exploit the Gensim library to train the Word2Vec, TF-IDF and Doc2Vec models. The input data format expected is in the form of text files containing sequences of system calls.

When designing the data processing pipelines, we utilized the behavioral patterns, considering that the communication between objects and in the data processing pipeline, and the input formats are the same. In particular, we used the strategy pattern by grouping the evaluated feature extraction algorithms into a single family of algorithms. We made sure that each algorithm from the Gensim library was encapsulated and had the same interface, so it could be interchanged without modifying the data processing pipeline. Similarly, we utilized the sci-kit learn library family of classification algorithms and were able to evaluate different combinations of algorithms with the tuning of their parameters, and feature extraction algorithms, to execute the whole pipeline without manual modifications. The features extracted by our implementations can be exploited by any machine learning method to perform intrusion detection as a binary classification task.

RESULTS

Competitor methods

Bag of system calls

Inspired by the study by Kang *et al.*[22], we enumerated the global set of system calls in the training data and adopted a key-value data structure. In this structure, a key corresponds to the combination of a trace ID and a system call, and the corresponding value represents the frequency of the system call in the trace. We used this data structure to generate the final dataset in matrix form.

Subsequence vector

Similarly to the aforementioned approach followed for Bag of System Calls, we enumerated the global set of system calls in the training data and generated a data matrix in which each row was a trace and each column was a system call. The entry in this matrix was initially calculated as the frequency of the system call. Subsequently, following the approach followed by Xie *et al.*[5], we re-calculated the entries in the matrix as the product between the system call and its frequency in each trace.

Table 1. Descriptive statistics for all datasets considered in this study

Dataset	Number of traces	Normal traces	Attack traces	Imbalance ratio
ADFA-LD[16]	5,951	5,205	746	6.98
NGIDS-DS[17]	37,377	19,256	18,121	1.06
WWW2019[18]	152,630	43,725	108,905	0.40

The reported imbalance ratio represents the proportion between the number of samples of the majority class and the number of samples of the minority class

Doc2Vec

The goal of Doc2Vec is to create a numeric representation of a document, regardless of its length. While word vectors represent the concept of a word, the document vector intends to represent the concept of a document. We propose this model as an alternative to Word2Vec for feature extraction applied directly to network traces.

Experimental setup

In our experiments, we assessed 5 feature extraction methods on 3 intrusion detection datasets. Descriptive statistics for all datasets considered in this study are reported in Table 1. For evaluation, we adopted a stratified 5-fold cross-validation scheme. The classification algorithm considered in our experiments was Extremely Randomized Trees (ERT), a state-of-the-art ensemble learning method based on decision trees. We emphasize that identifying the best machine learning algorithm is out of the scope of this paper. However, the features extracted with our method are general and, in principle, any machine learning algorithm can be used for the purpose of classification. Our aim was to show the potential of the features extracted using a conventional machine learning algorithm for classification.

For Word2Vec and Doc2Vec, we used a standard value for the embedding size ($k = 128$). For ERT, we used a standard configuration for the number of trees parameter ($T = 1000$). Since the datasets considered were imbalanced, we considered results in terms of macro precision, recall and F-score, to give the same importance to both classes in the average scores. We also report results in terms of area under the ROC curve (AUC). All the experimental results are reported in Table 2.

DISCUSSION

The results showed that word embedding-based feature extraction methods outperformed by a good margin all competitors with the NGIDS-DS dataset and the WWW2019 dataset. In these cases, the proposed variant of Word2Vec with TF-IDF weighting, appeared to obtain the best results. This behavior was not observed with the ADFA-LD dataset, where word embedding-based methods appear sub-optimal.

One possible explanation is that, when most of the system calls appearing in network traces are sparsely correlated, the semantic representation extracted by language models does not provide any advantage with respect to simpler frequency-based and pattern-based methods. On the contrary, the high-dimensionality of the new representation makes the classification task more difficult for the subsequent machine learning algorithm.

Another aspect that could disadvantage word embedding representations is that of the imbalance ratio between normal and attack traces. In fact, in the ADFA-LD dataset the imbalance ratio was 6.98, whereas the NGIDS-DS and WWW2019 datasets were more balanced, having an imbalance ratio of 1.06 and 0.40, respectively [Table 1]. This aspect is known to lead to increased challenges in classification tasks[36].

It is noteworthy that, among the word embedding-based methods, Doc2Vec performs poorly in all cases. This unexpected result shows that the preferred data granularity for traces in the context of intrusion

Table 2. Classification performance of extremely randomized trees models with different feature extraction techniques using different intrusion detection datasets

Dataset	Feature extraction technique	Precision (Macro)	Recall (Macro)	F-score (Macro)	Accuracy	AUC	F-score improvement over baseline (%)
ADFA-LD[16]	Bag of System Calls	0.9603	0.9244	**0.9414**	0.9752	0.9904	2.12%
	Subsequence Vector	0.9402	0.9053	0.9218	0.9670	0.9846	/
	Word2Vec	0.9376	0.8862	0.9096	0.9626	0.9791	-1.32%
	Word2Vec + TF-IDF	0.9246	0.8702	0.8948	0.9568	0.9762	-2.92%
	Doc2Vec	0.9006	0.5158	0.4985	0.8783	0.7457	-45.92%
NGIDS-DS[17]	Bag of System Calls	0.9689	0.9691	0.9690	0.9690	0.9937	1.38%
	Subsequence Vector	0.9557	0.9560	0.9558	0.9558	0.9899	/
	Word2Vec	0.9999	0.9999	0.9999	0.9999	0.9999	4.61%
	Word2Vec + TF-IDF	1.0000	1.0000	**1.0000**	1.0000	1.0000	4.62%
	Doc2Vec	0.7398	0.6560	0.6289	0.6648	0.7462	-34.20%
WWW2019[18]	Bag of System Calls	0.9568	0.9108	0.9303	0.9457	0.9823	13.68%
	Subsequence Vector	0.9830	0.8281	0.8183	0.9476	0.9048	/
	Word2Vec	0.9971	0.9929	0.9950	0.9959	0.9999	21.59%
	Word2Vec + TF-IDF	0.9990	0.9992	**0.9991**	0.9999	0.9999	22.09%
	Doc2Vec	0.8894	0.6478	0.6662	0.7981	0.7179	-18.58%

Results with three datasets: Australian Defence Force Academy Linux (ADFA-LD), Next-Generation Intrusion Detection System (NGIDS-DS), and Web Conference 2019 (WWW2019). Best results in terms of macro F-score are marked in bold. TF-IDF: Term-Frequency and Inverse Document Frequency

Table 3. Training and prediction execution time of the different feature extraction techniques using different intrusion detection datasets

Dataset	Feature extraction technique	Training time (min)	Prediction time (s)
ADFA-LD[16]	Bag of System Calls	0.13	0.25
	Subsequence Vector	0.15	0.28
	Word2Vec	1.95	0.36
	Word2Vec + TF-IDF	80.43	0.45
	Doc2Vec	0.88	0.30
NGIDS-DS[17]	Bag of System Calls	0.86	0.82
	Subsequence Vector	0.88	0.84
	Word2Vec	26.03	1.05
	Word2Vec + TF-IDF	1060.3	1.32
	Doc2Vec	4.05	0.88
WWW2019[18]	Bag of System Calls	1.23	1.18
	Subsequence Vector	1.21	1.16
	Word2Vec	18	1.45
	Word2Vec + TF-IDF	529.3	1.82
	Doc2Vec	26.08	1.22

Results with three datasets: Australian Defence Force Academy Linux (ADFA-LD), Next-Generation Intrusion Detection System (NGIDS-DS), and Web Conference 2019 (WWW2019). TF-IDF: Term-Frequency and Inverse Document Frequency

detection is that represented by system calls processed separately and aggregated using the compositionality property, rather than the whole trace represented directly as a vector.

In Table 3 we report the average execution times observed with the different feature extraction techniques. The execution was performed on a workstation equipped with an AMD Ryzen 5 1600 Processor (3200 MHz, 6 cores, 12 logical processors) with 32 GB of DDR4 RAM. The results show that frequency-based methods appear very efficient, even if they lead to sub-optimal results in terms of accuracy, as discussed before. More sophisticated feature extraction techniques are computationally more intensive, and in particular Word2Vec in combination with TF-IDF exhibits the highest execution time among all the techniques tested in this study. However, the leading time of the method is motivated by the training time of the TF-IDF dictionary which, in our experiments, is performed from scratch at every execution. In practice, there is the possibility to reduce this cost drastically by incrementally updating the TF-IDF model. Moreover, once models are trained and deployed, their prediction time appears similar for all of them, on the order of milliseconds. We argue that, in a production setting, training models from scratch is not required

continuously, but periodically, and it can be performed offline, while previously learned models are still active to perform intrusion detection. For these reasons, a higher accuracy in the predictive task is still important to pursue, since it can lead to the identification of complex attacks that would not be detected by simpler feature extraction techniques. Such attacks could have a significant negative impact on the organizations targeted by attackers. Considering the adoption of techniques with a higher computational cost can also be mitigated by designing parallel or high-performance computing implementations[23,24].

In conclusion, even if the results presented in this study are not vast enough to demonstrate the superiority of the proposed method on a broad scale, they are meant to show the potential of word embeddings to extract a new representation for network traces that can be used to carry out intrusion detection tasks accurately. Feature extraction based on word embedding models requires a higher computational time than simpler techniques, but leads to a higher accuracy, which is important for the identification of complex attacks. In future work, we aim to perform an extensive evaluation with different learning scenarios and machine learning algorithms. We also aim to study in detail word embedding representations and understand how to enforce them with more sophisticated data processing steps.

DECLARATIONS

Authors' contributions

Methodology, data acquisition, implementation, redaction of manuscript and analysis of experimental results: Corizzo R
Methodology, implementation and experiments: Zdravevski E
Implementation of feature extraction prototypes: Russell M, Vagliano A
Hypothesis formulation, methodology, data acquisition and analysis of experimental results: Japkowicz N

Availability of data and materials

Datasets are publicly available at the references reported in the Results section.

Financial support and sponsorship

We acknowledge the support of the Defense Advanced Research Projects Agency (DARPA) through the project "Lifelong Streaming Anomaly Detection" (Grant No. A19-0131-003).

Conflicts of interest

All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

REFERENCES

1. Axelsson S. Intrusion detection systems: a survey and taxonomy. Tech Rep 2000:99.
2. Bivens A, Palagiri C, Smith R, Szymanski B, Embrechts M. Network-based intrusion detection using neural networks. Intell Eng Syst Artif Neural Netw 2002;12:579-84.
3. Zhang J, Zulkernine M, Haque A. Random-forests-based network intrusion detection systems. IEEE Trans Syst Man Cybern C 2008;38:649-59.
4. Kruegel C, Toth T. Using decision trees to improve signature based intrusion detection. International Workshop on Recent Advances in

Intrusion Detection; 2003 Aug 8-10. Berlin: Springer; 2003. pp. 173-91.
5. Xie M, Hu J. Evaluating host-based anomaly detection systems: a preliminary analysis of ADFA-LD. 6th International Congress on Image and Signal Processing (CISP). Hangzhou, China; 2013. pp. 1711-6.
6. Xie M, Hu J, Yu X, Chang E. Evaluating host-based anomaly detection systems: Application of the frequency-based algorithms to ADFA-LD. International Conference on Network and System Security. Springer, Cham; 2015. pp. 542-9.
7. Aghaei E, Serpen G. Ensemble classifier for misuse detection using N-gram feature vectors through operating system call traces. Int J Hybrid Intell Syst 2017;14:141-54.
8. Ahmim A, Derdour M, Ferrag MA. An intrusion detection system based on combining probability predictions of a tree of classifiers. Int J Commun Syst 2018;31:e3547.1-17.
9. Wunderlich S, Ring M, Landes D, Hotho A. Comparison of system call representations for intrusion detection. International Joint Conference: 12th International Conference on Computational Intelligence in Security for Information Systems (CISIS 2019) and 10th International Conference on EUropean Transnational Education (ICEUTE 2019); 2019. Seville, Spain; 2019. pp. 14-24.
10. Hofmeyr SA, Forrest S, Somayaji A. Intrusion detection using sequences of system calls. J Comput Secur 1998;6:151-80.
11. Lippmann R. DARPA Intrusion Detection Data Sets. Available from: https://www.ll.mit.edu/r-d/datasets. [Last accessed on 31 Jul 2020]
12. Hettich S, Bay S. KDD Cup 1999 Dataset. Available from: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html. [Last accessed on 31 Jul 2020]
13. Amiri F, Yousefi MR, Lucas C, Shakery A, Shakery A. Mutual information-based feature selection for intrusion detection systems. J Netw Comput Appl 2011;34:1184-99.
14. Brahmi H, Brahmi I, Ben Yahia SB. OMC-IDS: at the cross-roads of OLAP mining and intrusion detection. Advances in Knowledge Discovery and Data Mining:16th Pacific-Asia Conference; 2012 May 29-June 1; Kuala Lumpur, Malaysia: Verlag, Springer; 2012. pp. 13-24.
15. Apiletti D, Baralis E, Cerquitelli T, D'Elia V. Characterizing network traffic by means of the NetMine framework. Comput Netw 2009;53:774-89.
16. Bilge L, Balzarotti D, Robertson W, Kirda E, Kruegel C. Disclosure: detecting botnet command and control servers through large-scale netflow analysis leyla. ACSAC '12: Proceedings of the 28th Annual Computer Security Applications Conference; 2012 Dec. Orlando, Florida, USA; 2012. pp. 129-38.
17. Hu J. The ADFA intrusion detection datasets. Available from: https://www.unsw.adfa.edu.au/australian-centre-for-cyber-security/cybersecurity/ADFA-IDS-Datasets/. [Last accessed on 31 Jul 2020]
18. Creech G, Hu J. Generation of a new IDS test dataset: time to retire the KDD collection. 2013 IEEE Wireless Communications and Networking Conference (WCNC); 2002. Shanghai, China; 2013. pp. 4487-92.
19. Haider W, Hu J, Slay J, Turnbull BP, Xie Y. Generating realistic intrusion detection system dataset based on fuzzy qualitative modeling. J Netw Comput Appl 2017;87:185-92.
20. Li YF, Gao Y, Ayoade G, Tao H, Khan L, et al. Multistream classification for cyber threat data with heterogeneous feature space. The World Wide Web Conference, 2019 May. San Francisco, USA; 2019. pp. 2992-8.
21. Liu Z, Japkowicz N, Wang R, Cai Y, Tang D, et al. A statistical pattern based feature extraction method on system call traces for anomaly detection. Inform Software Tech 2020;126:106348.
22. Kang DK, Fuller D, Honavar V. Learning classifiers for misuse and anomaly detection using a bag of system calls representation. Proceedings from the 6th Annual IEEE System, Man and Cybernetics Information Assurance Workshop; 2005 June 15th-17th. West Point, NY, USA; 2005. pp. 118-25.
23. Corizzo R, Ceci M, Japkowicz N. Anomaly detection and repair for accurate predictions in geo-distributed big data. Big Data Res 2019;16:18-35.
24. Corizzo R, Ceci M, Zdravevski E, Japkowicz N. Scalable auto-encoders for gravitational waves detection from time series data. Expert Syst Appl 2020;151:113378.
25. Mikolov T, Sutskever I, Chen K, Corrado G, Dean J. Distributed representations of words and phrases and their compositionality. NIPS'13: Proceedings of the 26th International Conference on Neural Information Processing Systems; 2013 Dec. Lake Tahoe, USA; 2013. pp. 3111-9. arXiv1310.4546 [Preprint]. October 16, 2013. Available from: https://arxiv.org/abs/1310.4546. [Last accessed on 31 Jul 2020]
26. Globerson A, Gal C, Fernando P, Naftali T. Euclidean embedding of co-occurrence data. J Mach Learn Res 2007;8:2265-95.
27. Levy O, Goldberg Y. Neural word embedding as implicit matrix factorization. NIPS'14: Proceedings of the 27th International Conference on Neural Information Processing Systems; 2014 Dec. Montreal, Canada; 2014. pp. 2177-85.
28. Lavelli A, Sebastiani F, Zanoli R. Distributional term representations: an experimental comparison. CIKM '04: Proceedings of the thirteenth ACM international conference on Information and knowledge management; 2004 Nov. Washington D.C, USA; pp. 615-24.
29. Firth JR. A synopsis of linguistic theory, 1930-1955. Studies in Linguistics Analysis. Oxford: Philological Society; 1957. pp. 1-32.
30. Mikolov T, Chen K, Corrado G, Dean J. Efficient estimation of word representations in vector space. arXiv 1310.3781 [Preprint] September 7, 2013. Available from: https://arxiv.org/abs/1301.3781. [Last accessed on 31 Jul 2020]
31. Jang B, Kim I, Kim JW. Word2vec convolutional neural networks for classification of news articles and tweets. PLoS One 2019;14:e0220976.
32. Wu C, Gao R, Zhang Y, De Marinis Y. PTPD: predicting therapeutic peptides by deep learning and word2vec. BMC Bioinformatics 2019;20:456.
33. Chen T, Mao Q, Lv M, Cheng H, Li Y. DroidVecDeep: android malware detection based on word2vec and deep belief network. TIIS

2019;13:2180-97.
34. Baek JW, Chung KY. Multimedia recommendation using word2vec-based social relationship mining. Multimed Tools Appl 2020:1-17.
35. Singhal A. Modern information retrieval: a brief overview. IEEE Data Eng Bull 2001;24:35-43.
36. Branco P, Torgo L, Ribeiro RP. A survey of predictive modeling on imbalanced domains. ACM Computing Surveys 2016;49:1-50.

AUTHOR INSTRUCTIONS

1. Submission Overview

Before you decide to publish with us, please read the following items carefully and make sure that you are well aware of Editorial Policies and the following requirements.

1.1 Topic Suitability

The topic of the manuscript must fit the scope of the journal. Please refer to Aims and Scope for more information.

1.2 Open Access and Copyright

The journal adopts Gold Open Access publishing model since its establishment and has been distributing contents under Attribution 4.0 International License since October 2017, whereas Attribution-NonCommercial-ShareAlike 3.0 Unported had been adopted by then. Please make sure that you are well aware of these policies.

1.3 Publication Fees

Authors are required to pay Article Processing Charges of 360 US Dollars after the manuscript is officially accepted. For more details, please refer to Article Processing Charges.

1.4 Language Editing

All submissions are required to be presented clearly and cohesively in good English. Authors whose first language is not English are advised to have their manuscripts checked or edited by a native English speaker before submission to ensure the high quality of expression. A well-organized manuscript in good English would make the peer review even the whole editorial handling more smooth and efficient.

If needed, authors are recommended to consider the language editing services provided by Charlesworth to ensure that the manuscript is written in correct scientific English before submission. Authors who publish with OAE journals enjoy a special discount for the services of Charlesworth via the following two ways.

Submit your manuscripts directly at http://www.charlesworthauthorservices.com/~OAE;

Open the link http://www.charlesworthauthorservices.com/, and enter Promotion Code “OAE” when you submit.

1.5 Work Funded by the National Institutes of Health

If an accepted manuscript was funded by National Institutes of Health (NIH), the author may inform editors of the NIH funding number. The editors are able to deposit the paper to the NIH Manuscript Submission System on behalf of the author.

2. Submission Preparation

2.1 Cover Letter

A cover letter is required to be submitted accompanying each manuscript. It should be concise and explain why the study is significant, why it fits the scope of the journal, and why it would be attractive to readers, *etc.*

Here is a guideline of a cover letter for authors’ consideration:

In the first paragraph: include the title and type (e.g., Original Article, Review, Case Report, *etc.*) of the manuscript, a brief on the background of the study, the question the author sought out to answer and why;

In the second paragraph: concisely explain what was done, the main findings and why they are significant;

In the third paragraph: indicate why the manuscript fits the Aims and Scope of the journal, and why it would be attractive to readers;

In the fourth paragraph: confirm that the manuscript has not been published elsewhere and not under consideration of any other journal. All authors have approved the manuscript and agreed on its submission to the journal. Journal’s specific requirements have been met if any.

If the manuscript is contributed to a special issue, please also mention it in the cover letter.

If the manuscript was presented partly or entirely in a conference, the author should clearly state the background information of the event, including the conference name, time and place in the cover letter.

2.2 Types of Manuscripts

There is no restriction on the length of manuscripts, number of figures, tables and references, provided that the manuscript is concise and comprehensive. The journal publishes Original Article, Review, Meta-Analysis, Case Report, Commentary, *etc*. For more details about paper type, please refer to the following table.

Manuscript Type	Definition	Abstract	Keywords	Main Text Structure
Original Article	An Original Article describes detailed results from novel research. All findings are extensively discussed.	Structured abstract including Aim, Methods, Results and Conclusion. No more than 250 words.	3-8 keywords	The main content should include four sections: Introduction, Methods, Results and Discussion.
Review	A Review paper summarizes the literature on previous studies. It usually does not present any new information on a subject.	Unstructured abstract. No more than 250 words.	3-8 keywords	The main text may consist of several sections with unfixed section titles. We suggest that the author includes an "Introduction" section at the beginning, several sections with unfixed titles in the middle part, and a "Conclusion" section in the end.
Case Report	A Case Report details symptoms, signs, diagnosis, treatment, and follows up an individual patient. The goal of a Case Report is to make other researchers aware of the possibility that a specific phenomenon might occur.	Unstructured abstract. No more than 150 words.	3-8 keywords	The main text consists of three sections with fixed section titles: Introduction, Case Report, and Discussion.
Meta-Analysis	A Meta-Analysis is a statistical analysis combining the results of multiple scientific studies. It is often an overview of clinical trials.	Structured abstract including Aim, Methods, Results and Conclusion. No more than 250 words.	3-8 keywords	The main content should include four sections: Introduction, Methods, Results and Discussion.
Systematic Review	A Systematic Review collects and critically analyzes multiple research studies, using methods selected before one or more research questions are formulated, and then finding and analyzing related studies and answering those questions in a structured methodology.	Structured abstract including Aim, Methods, Results and Conclusion. No more than 250 words.	3-8 keywords	The main content should include four sections: Introduction, Methods, Results and Discussion.
Technical Note	A Technical Note is a short article giving a brief description of a specific development, technique or procedure, or it may describe a modification of an existing technique, procedure or device applied in research.	Unstructured abstract. No more than 250 words.	3-8 keywords	/
Commentary	A Commentary is to provide comments on a newly published article or an alternative viewpoint on a certain topic.	Unstructured abstract. No more than 250 words.	3-8 keywords	/
Editorial	An Editorial is a short article describing news about the journal or opinions of senior editors or the publisher.	None required	None required	/
Letter to Editor	A Letter to Editor is usually an open post-publication review of a paper from its readers, often critical of some aspect of a published paper. Controversial papers often attract numerous Letters to Editor	Unstructured abstract (optional). No more than 250 words.	3-8 keywords (optional)	/
Opinion	An Opinion usually presents personal thoughts, beliefs, or feelings on a topic.	Unstructured abstract (optional). No more than 250 words.	3-8 keywords	/
Perspective	A Perspective provides personal points of view on the state-of-the-art of a specific area of knowledge and its future prospects. Links to areas of intense current research focus can also be made. The emphasis should be on a personal assessment rather than a comprehensive, critical review. However, comments should be put into the context of existing literature. Perspectives are usually invited by the Editors.	Unstructured abstract. No more than 150 words.	3-8 keywords	/

2.3 Manuscript Structure

2.3.1 Front Matter

2.3.1.1 Title

The title of the manuscript should be concise, specific and relevant, with no more than 16 words if possible. When gene or protein names are included, the abbreviated name rather than full name should be used.

2.3.1.2 Authors and Affiliations

Authors' full names should be listed. The initials of middle names can be provided. Institutional addresses and email addresses for all authors should be listed. At least one author should be designated as corresponding author. In addition, corresponding authors are suggested to provide their Open Researcher and Contributor ID upon submission. Please note that any change to authorship is not allowed after manuscript acceptance.

2.3.1.3 Abstract

The abstract should be a single paragraph with word limitation and specific structure requirements (for more details please refer to Types of Manuscripts). It usually describes the main objective(s) of the study, explains how the study was done, including any model organisms used, without methodological detail, and summarizes the most important results and their significance. The abstract must be an objective representation of the study: it is not allowed to contain results which are not presented and substantiated in the manuscript, or exaggerate the main conclusions. Citations should not be included in the abstract.

2.3.1.4 Keywords

Three to eight keywords should be provided, which are specific to the article, yet reasonably common within the subject discipline.

2.3.2 Main Text

Manuscripts of different types are structured with different sections of content. Please refer to Types of Manuscripts to make sure which sections should be included in the manuscripts.

2.3.2.1 Introduction

The introduction should contain background that puts the manuscript into context, allow readers to understand why the study is important, include a brief review of key literature, and conclude with a brief statement of the overall aim of the work and a comment about whether that aim was achieved. Relevant controversies or disagreements in the field should be introduced as well.

2.3.2.2 Methods

Methods should contain sufficient details to allow others to fully replicate the study. New methods and protocols should be described in detail while well-established methods can be briefly described or appropriately cited. Experimental participants selected, the drugs and chemicals used, the statistical methods taken, and the computer software used should be identified precisely. Statistical terms, abbreviations, and all symbols used should be defined clearly. Protocol documents for clinical trials, observational studies, and other non-laboratory investigations may be uploaded as supplementary materials.

2.3.2.3 Results

This section contains the findings of the study. Results of statistical analysis should also be included either as text or as tables or figures if appropriate. Authors should emphasize and summarize only the most important observations. Data on all primary and secondary outcomes identified in the section Methods should also be provided. Extra or supplementary materials and technical details can be placed in supplementary documents.

2.3.2.4 Discussion

This section should discuss the implications of the findings in context of existing research and highlight limitations of the study. Future research directions may also be mentioned.

2.3.2.5 Conclusion

It should state clearly the main conclusions and include the explanation of their relevance or importance to the field.

2.3.3 Back Matter

2.3.3.1 Acknowledgments

Anyone who contributed towards the article but does not meet the criteria for authorship, including those who provided professional writing services or materials, should be acknowledged. Authors should obtain permission to acknowledge from all those mentioned in the Acknowledgments section. This section is not added if the author does not have anyone to acknowledge.

2.3.3.2 Authors' Contributions

Each author is expected to have made substantial contributions to the conception or design of the work, or the acquisition, analysis, or interpretation of data, or the creation of new software used in the work, or have drafted the work or substantively revised it.

Please use Surname and Initial of Forename to refer to an author's contribution. For example: made substantial contributions to conception and design of the study and performed data analysis and interpretation: Salas H, Castaneda WV; performed data acquisition, as well as provided administrative, technical, and material support: Castillo N, Young V.

If an article is single-authored, please include "The author contributed solely to the article." in this section.

2.3.3.3 Availability of Data and Materials

In order to maintain the integrity, transparency and reproducibility of research records, authors should include this section in their manuscripts, detailing where the data supporting their findings can be found. Data can be deposited into data repositories or published as supplementary information in the journal. Authors who cannot share their data should state that the data will not be shared and explain it. If a manuscript does not involve such issue, please state "Not applicable." in this section.

2.3.3.4 Financial Support and Sponsorship

All sources of funding for the study reported should be declared. The role of the funding body in the experiment design, collection, analysis and interpretation of data, and writing of the manuscript should be declared. Any relevant grant numbers and the link of funder's website should be provided if any. If the study is not involved with this issue, state "None." in this section.

2.3.3.5 Conflicts of Interest

Authors must declare any potential conflicts of interest that may be perceived as inappropriately influencing the representation or interpretation of reported research results. If there are no conflicts of interest, please state "All authors declared that there are no conflicts of interest." in this section. Some authors may be bound by confidentiality agreements. In such cases, in place of itemized disclosures, we will require authors to state "All authors declare that they are bound by confidentiality agreements that prevent them from disclosing their conflicts of interest in this work.". If authors are unsure whether conflicts of interest exist, please refer to the "Conflicts of Interest" of OAE Editorial Policies for a full explanation.

2.3.3.6 Ethical Approval and Consent to Participate

Research involving human subjects, human material or human data must be performed in accordance with the Declaration of Helsinki and approved by an appropriate ethics committee. An informed consent to participate in the study should also be obtained from participants, or their parents or legal guardians for children under 16. A statement detailing the name of the ethics committee (including the reference number where appropriate) and the informed consent obtained must appear in the manuscripts reporting such research.

Studies involving animals and cell lines must include a statement on ethical approval. More information is available at Editorial Policies.

If the manuscript does not involve such issue, please state "Not applicable." in this section.

2.3.3.7 Consent for Publication

Manuscripts containing individual details, images or videos, must obtain consent for publication from that person, or in the case of children, their parents or legal guardians. If the person has died, consent for publication must be obtained from the next of kin of the participant. Manuscripts must include a statement that a written informed consent for publication was obtained. Authors do not have to submit such content accompanying the manuscript. However, these documents must be available if requested. If the manuscript does not involve this issue, state "Not applicable." in this section.

2.3.3.8 Copyright

Authors retain copyright of their works through a Creative Commons Attribution 4.0 International License that clearly states how readers can copy, distribute, and use their attributed research, free of charge. A declaration "© The Author(s) 2020." will be added to each article. Authors are required to sign License to Publish before formal publication.

2.3.3.9 References

References should be numbered in order of appearance at the end of manuscripts. In the text, reference numbers should be placed in square brackets and the corresponding references are cited thereafter. Only the first five authors' names are required to be listed in the references, other authors' names should be omitted and replaced with "et al.". Abbreviations of the journals should be provided on the basis of Index Medicus. Information from manuscripts accepted but not published should be cited in the text as "Unpublished material" with written permission from the source.

References should be described as follows, depending on the types of works:

Types	Examples
Journal articles by individual authors	Weaver DL, Ashikaga T, Krag DN, Skelly JM, Anderson SJ, et al. Effect of occult metastases on survival in node-negative breast cancer. N Engl J Med 2011;364:412-21. [PMID: 21247310 DOI: 10.1056/NEJMoa1008108]
Organization as author	Diabetes Prevention Program Research Group. Hypertension, insulin, and proinsulin in participants with impaired glucose tolerance. Hypertension 2002;40:679-86. [PMID: 12411462]
Both personal authors and organization as author	Vallancien G, Emberton M, Harving N, van Moorselaar RJ; Alf-One Study Group. Sexual dysfunction in 1,274 European men suffering from lower urinary tract symptoms. J Urol 2003;169:2257-61. [PMID: 12771764 DOI: 10.1097/01.ju.0000067940.76090.73]
Journal articles not in English	Zhang X, Xiong H, Ji TY, Zhang YH, Wang Y. Case report of anti-N-methyl-D-aspartate receptor encephalitis in child. J Appl Clin Pediatr 2012;27:1903-7. (in Chinese)
Journal articles ahead of print	Odibo AO. Falling stillbirth and neonatal mortality rates in twin gestation: not a reason for complacency. BJOG 2018; Epub ahead of print [PMID: 30461178 DOI: 10.1111/1471-0528.15541]
Books	Sherlock S, Dooley J. Diseases of the liver and billiary system. 9th ed. Oxford: Blackwell Sci Pub; 1993. pp. 258-96.
Book chapters	Meltzer PS, Kallioniemi A, Trent JM. Chromosome alterations in human solid tumors. In: Vogelstein B, Kinzler KW, editors. The genetic basis of human cancer. New York: McGraw-Hill; 2002. pp. 93-113.
Online resource	FDA News Release. FDA approval brings first gene therapy to the United States. Available from: https://www.fda.gov/NewsEvents/Newsroom/PressAnnouncements/ucm574058.htm. [Last accessed on 30 Oct 2017]
Conference proceedings	Harnden P, Joffe JK, Jones WG, editors. Germ cell tumours V. Proceedings of the 5th Germ Cell Tumour Conference; 2001 Sep 13-15; Leeds, UK. New York: Springer; 2002.
Conference paper	Christensen S, Oppacher F. An analysis of Koza's computational effort statistic for genetic programming. In: Foster JA, Lutton E, Miller J, Ryan C, Tettamanzi AG, editors. Genetic programming. EuroGP 2002: Proceedings of the 5th European Conference on Genetic Programming; 2002 Apr 3-5; Kinsdale, Ireland. Berlin: Springer; 2002. pp. 182-91.
Unpublished material	Tian D, Araki H, Stahl E, Bergelson J, Kreitman M. Signature of balancing selection in Arabidopsis. Proc Natl Acad Sci U S A. Forthcoming 2002.

For other types of references, please refer to U.S. National Library of Medicine.
The journal also recommends that authors prepare references with a bibliography software package, such as EndNote to avoid typing mistakes and duplicated references.

2.3.3.10 Supplementary Materials

Additional data and information can be uploaded as Supplementary Material to accompany the manuscripts. The supplementary materials will also be available to the referees as part of the peer-review process. Any file format is acceptable, such as data sheet (word, excel, csv, cdx, fasta, pdf or zip files), presentation (powerpoint, pdf or zip files), image (cdx, eps, jpeg, pdf, png or tiff), table (word, excel, csv or pdf), audio (mp3, wav or wma) or video (avi, divx, flv, mov, mp4, mpeg, mpg or wmv). All information should be clearly presented. Supplementary materials should be cited in the main text in numeric order (e.g., Supplementary Figure 1, Supplementary Figure 2, Supplementary Table 1, Supplementary Table 2, *etc.*). The style of supplementary figures or tables complies with the same requirements on figures or tables in main text. Videos and audios should be prepared in English, and limited to a size of 500 MB or a duration of 3 minutes.

2.4 Manuscript Format

2.4.1 File Format

Manuscript files can be in DOC and DOCX formats and should not be locked or protected.

2.4.2 Length

There are no restrictions on paper length, number of figures, or amount of supporting documents. Authors are encouraged to present and discuss their findings concisely.

2.4.3 Language

Manuscripts must be written in English.

2.4.4 Multimedia Files

The journal supports manuscripts with multimedia files. The requirements are listed as follows:
Videos or audio files are only acceptable in English. The presentation and introduction should be easy to understand. The frames should be clear, and the speech speed should be moderate.
A brief overview of the video or audio files should be given in the manuscript text.
The video or audio files should be limited to a duration of 3 min and a size of up to 500 MB.

Please use professional software to produce high-quality video files, to facilitate acceptance and publication along with the submitted article. Upload the videos in mp4, wmv, or rm format (preferably mp4) and audio files in mp3 or wav format.

2.4.5 Figures

Figures should be cited in numeric order (e.g., Figure 1, Figure 2) and placed after the paragraph where it is first cited;
Figures can be submitted in format of tiff, psd, AI or jpeg, with resolution of 300-600 dpi;
Figure caption is placed under the Figure;
Diagrams with describing words (including, flow chart, coordinate diagram, bar chart, line chart, and scatter diagram, *etc*.) should be editable in word, excel or powerpoint format. Non-English information should be avoided;
Labels, numbers, letters, arrows, and symbols in figure should be clear, of uniform size, and contrast with the background;
Symbols, arrows, numbers, or letters used to identify parts of the illustrations must be identified and explained in the legend;
Internal scale (magnification) should be explained and the staining method in photomicrographs should be identified;
All non-standard abbreviations should be explained in the legend;
Permission for use of copyrighted materials from other sources, including re-published, adapted, modified, or partial figures and images from the internet, must be obtained. It is authors' responsibility to acquire the licenses, to follow any citation instruction requested by third-party rights holders, and cover any supplementary charges.

2.4.6 Tables

Tables should be cited in numeric order and placed after the paragraph where it is first cited;
The table caption should be placed above the table and labeled sequentially (e.g., Table 1, Table 2);
Tables should be provided in editable form like DOC or DOCX format (picture is not allowed);
Abbreviations and symbols used in table should be explained in footnote;
Explanatory matter should also be placed in footnotes;
Permission for use of copyrighted materials from other sources, including re-published, adapted, modified, or partial tables from the internet, must be obtained. It is authors' responsibility to acquire the licenses, to follow any citation instruction requested by third-party rights holders, and cover any supplementary charges.

2.4.7 Abbreviations

Abbreviations should be defined upon first appearance in the abstract, main text, and in figure or table captions and used consistently thereafter. Non-standard abbreviations are not allowed unless they appear at least three times in the text. Commonly-used abbreviations, such as DNA, RNA, ATP, *etc*., can be used directly without definition. Abbreviations in titles and keywords should be avoided, except for the ones which are widely used.

2.4.8 Italics

General italic words like *vs*., *et al., etc*., *in vivo*, *in vitro*; *t* test, *F* test, *U* test; related coefficient as *r*, sample number as *n*, and probability as *P*; names of genes; names of bacteria and biology species in Latin.

2.4.9 Units

SI Units should be used. Imperial, US customary and other units should be converted to SI units whenever possible. There is a space between the number and the unit (i.e., 23 mL). Hour, minute, second should be written as h, min, s.

2.4.10 Numbers

Numbers appearing at the beginning of sentences should be expressed in English. When there are two or more numbers in a paragraph, they should be expressed as Arabic numerals; when there is only one number in a paragraph, number < 10 should be expressed in English and number > 10 should be expressed as Arabic numerals. 12345678 should be written as 12,345,678.

2.4.11 Equations

Equations should be editable and not appear in a picture format. Authors are advised to use either the Microsoft Equation Editor or the MathType for display and inline equations.

2.5 Submission Link

Submit an article via http://www.oaemesas.com/jsss.

OAE Publishing Inc.
www.oaepublish.com

Journal of Surveillance, Security and Safety
(*JSSS*)
Los Angeles Office
245 E Main Street ste122, Alhambra,
CA 91801, USA
Tel: +1 323 9987086
E-mail: editorialoffice@jsssjournal.com
Website: www.jsssjournal.com