International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

X.1366

(09/2020)

SERIES X: DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

Secure applications and services (2) – Internet of things (IoT) security

Aggregate message authentication schemes for Internet of things environment

Recommendation ITU-T X.1366

International Telecommunication Union

ITU-T X-SERIES RECOMMENDATIONS

DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

PUBLIC DATA NETWORKS	X.1–X.199
OPEN SYSTEMS INTERCONNECTION	X.200–X.299
INTERWORKING BETWEEN NETWORKS	X.300–X.399
MESSAGE HANDLING SYSTEMS	X.400–X.499
DIRECTORY	X.500–X.599
OSI NETWORKING AND SYSTEM ASPECTS	X.600–X.699
OSI MANAGEMENT	X.700–X.799
SECURITY	X.800–X.849
OSI APPLICATIONS	X.850–X.899
OPEN DISTRIBUTED PROCESSING	X.900–X.999
INFORMATION AND NETWORK SECURITY	
General security aspects	X.1000–X.1029
Network security	X.1030–X.1049
Security management	X.1050–X.1069
Telebiometrics	X.1080–X.1099
SECURE APPLICATIONS AND SERVICES (1)	
Multicast security	X.1100–X.1109
Home network security	X.1110–X.1119
Mobile security	X.1120–X.1139
Web security	X.1140–X.1149
Security protocols (1)	X.1150–X.1159
Peer-to-peer security	X.1160–X.1169
Networked ID security	X.1170–X.1179
IPTV security	X.1180–X.1199
CYBERSPACE SECURITY	
Cybersecurity	X.1200–X.1229
Countering spam	X.1230–X.1249
Identity management	X.1250–X.1279
SECURE APPLICATIONS AND SERVICES (2)	
Emergency communications	X.1300–X.1309
Ubiquitous sensor network security	X.1310–X.1319
Smart grid security	X.1330–X.1339
Certified mail	X.1340–X.1349
Internet of things (IoT) security	**X.1360–X.1369**
Intelligent transportation system (ITS) security	X.1370–X.1389
Distributed ledger technology security	X.1400–X.1429
Distributed ledger technology security	X.1430–X.1449
Security protocols (2)	X.1450–X.1459
CYBERSECURITY INFORMATION EXCHANGE	
Overview of cybersecurity	X.1500–X.1519
Vulnerability/state exchange	X.1520–X.1539
Event/incident/heuristics exchange	X.1540–X.1549
Exchange of policies	X.1550–X.1559
Heuristics and information request	X.1560–X.1569
Identification and discovery	X.1570–X.1579
Assured exchange	X.1580–X.1589
CLOUD COMPUTING SECURITY	
Overview of cloud computing security	X.1600–X.1601
Cloud computing security design	X.1602–X.1639
Cloud computing security best practices and guidelines	X.1640–X.1659
Cloud computing security implementation	X.1660–X.1679
Other cloud computing security	X.1680–X.1699
QUANTUM COMMUNICATION	
Terminologies	X.1700–X.1701
Quantum random number generator	X.1702–X.1709
Framework of QKDN security	X.1710–X.1711
Security design for QKDN	X.1712–X.1719
Security techniques for QKDN	X.1720–X.1729
DATA SECURITY	
Big Data Security	X.1750–X.1759
5G SECURITY	X.1800–X.1819

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T X.1366

Aggregate message authentication schemes for Internet of things environment

Summary

The number of Internet of things (IoT) devices is increasing, and in the near future there will be an enormous number of devices connected to the IoT network including 5G. Recommendation ITU-T X.1366 specifies two message authentication schemes. One is an aggregate message authentication (AMA) scheme for IoT as a basic mechanism. The other is an interactive aggregate message authentication (IAMA) scheme with interactive protocol in a lightweight and secure manner. Both aggregate message authentication schemes can be applied for ensuring "entity (identity) authentication" as well as for ensuring "message authentication". These schemes may not be applicable in all use cases for utilizing IoT devices, but they are quite effective and suitable for use cases in the following conditions where:

- Message authentication is required from tens to tens of thousands of IoT devices.
- Data or message being handled for an authentication process that occurs frequently and intermittently.

For example, "surveillance applications for use of image data" and "remote telemetry" such as monitoring of plant or factory operations and health monitoring are the typical candidates of use cases for these schemes.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T X.1366	2020-09-03	17	11.1002/1000/14262

Keywords

Aggregate message authentication, AMA, IoT.

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Recommendation ITU-T X.1366

Aggregate message authentication schemes for Internet of things environment

1 Scope

This Recommendation specifies two message authentication schemes. One is an aggregate message authentication (AMA) scheme for IoT as a basic mechanism. The other is an interactive aggregate message authentication (IAMA) scheme with interactive protocol in a lightweight and secure manner. Both aggregate message authentication schemes can be applied for ensuring "entity (identity) authentication" as well as for ensuring "message authentication".

How to implement these schemes in a specific IoT environment, as well as aggregate signature technologies are outside the scope of this Recommendation.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

None.

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following term defined elsewhere:

3.1.1 message authentication code (MAC) [b-ITU-T X.813]: A cryptographic check value that is used to provide data origin authentication and data integrity.

3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.2.1 message authentication: A property that guarantees that a message has not been modified while in transit to ensure data integrity, and allows the receiving party to verify the source of the message.

3.2.2 aggregate message authentication (AMA): A property that allows multiple message authentication codes, generated by multiple senders, to be aggregated into a shorter authentication code that can still be verified by a recipient who has the secret keys of the senders.

3.2.3 authentication tags: A piece of data to be used for message authentication.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

AGT Adaptive Group Testing protocol

AMA Aggregate Message Authentication

AMAC Aggregate Message Authentication Code

IAMA	Interactive Aggregate Message Authentication
IoT	Internet of Things
MAC	Message Authentication Code
XOR	Exclusive OR operation

5 Conventions

None.

6 Overview and basic concept

6.1 Overview

The number of Internet of things (IoT) devices is continually increasing, and in the near future there will be an enormous number of devices connected to the IoT network including 5G. This Recommendation provides a lightweight and secure authentication system that can be applied in such a situation.

Message authentication code (MAC) is one of the most fundamental cryptographic primitives, and the MAC can be used as a lightweight cryptographic primitive for message authentication. However, as shown in Figure 1, in current IoT systems, for messages sent from IoT devices, authentication tags (see clause 3.2.3) are individually generated at the IoT device side, and each message with a generated tag is basically verified by a verification process on the receiver side. The major problem recognized in this current IoT scenario is that the load of existing authentication and verification processes is increasing in proportion to the increase in the number of IoT devices.

Aggregate message authentication code (AMAC) is an existing technology that allows for the compression of multiple MAC tags on multiple messages generated by different devices into a single aggregate tag without compromising security (see Appendix II). The advantage of AMAC lies in the fact that the size of an aggregate tag is much smaller than the combined total sizes of MAC tags, and hence it will be useful in applications in mobile networks or IoT networks where many devices sending messages are connected. Specifically, AMAC can be used in applications to make networks using MACs more efficient. However, this method cannot identify invalid messages among the multiple messages once these messages are regarded as invalid using an aggregate-tag in AMAC in general. In this Recommendation, the existing AMAC scheme is extended so that it allows to compress multiple MAC tags with detection capability to specify invalid messages.

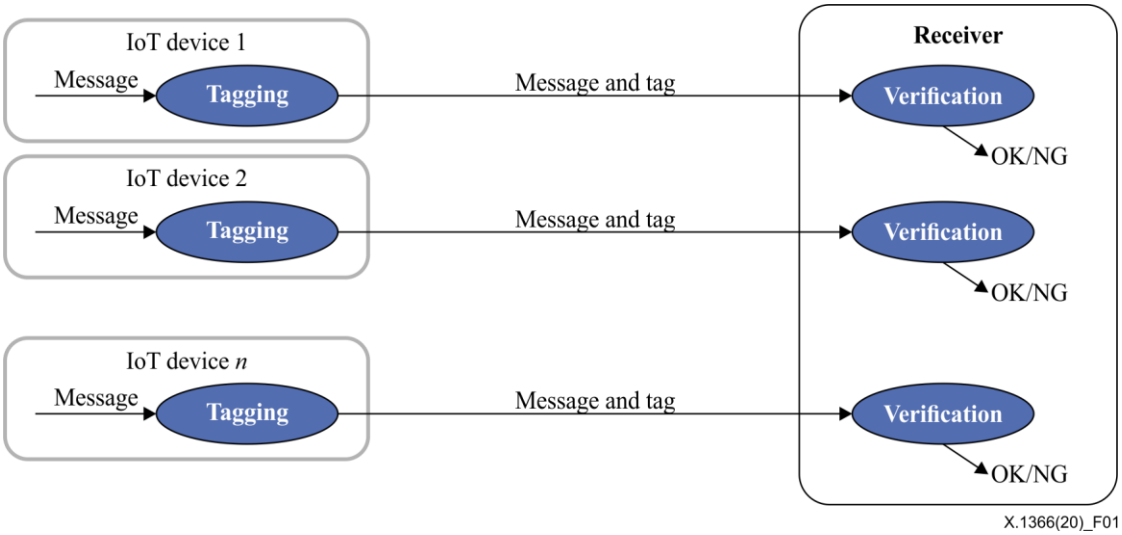

Figure 1 – One-to-one authentication system (conventional system)

6.2 Basic concept of aggregate message authentication system

6.2.1 General

Figure 2 shows the basic scheme of aggregate message authentication (AMA) proposed in this Recommendation. An aggregation node is installed into an IoT network system for aggregating MAC tags/authentication tags without changing the input-formats or structures of the existing MACs in the network. The aggregation node compresses multiple MAC tags attached in multiple messages generated by different devices into a single aggregate tag without compromising security, and the aggregate tag is transmitted through a main channel to a receiver to conduct verification processes for the tag. The receiver checks the validity of multiple messages by using the aggregate tag and can identify invalid messages or data from the aggregate tag. This technique is effective in reducing the volume of data transmitted when the size of the aggregate tag is much smaller than the total size of the multiple MAC tags.

This Recommendation describes an AMA scheme for IoT as a basic mechanism and an interactive AMA (IAMA) scheme to explain how the aggregation and verification processes are performed. In the AMA scheme, only the main channel from the aggregation node to the receiver is used to transmit the aggregate tag. The aggregation and verification algorithms of the AMA scheme is specified in clause 7. In the IAMA scheme, a feedback channel that is an authenticated channel with low bandwidth from the receiver to the aggregation node is also used in addition to the main channel. By transmitting a verification result from the receiver to the aggregation node through the feedback channel, the aggregation node can compress the MAC tags more effectively than the AMA in clause 7. An interactive protocol between the aggregation node and receiver is executed for verification as specified in clause 8.

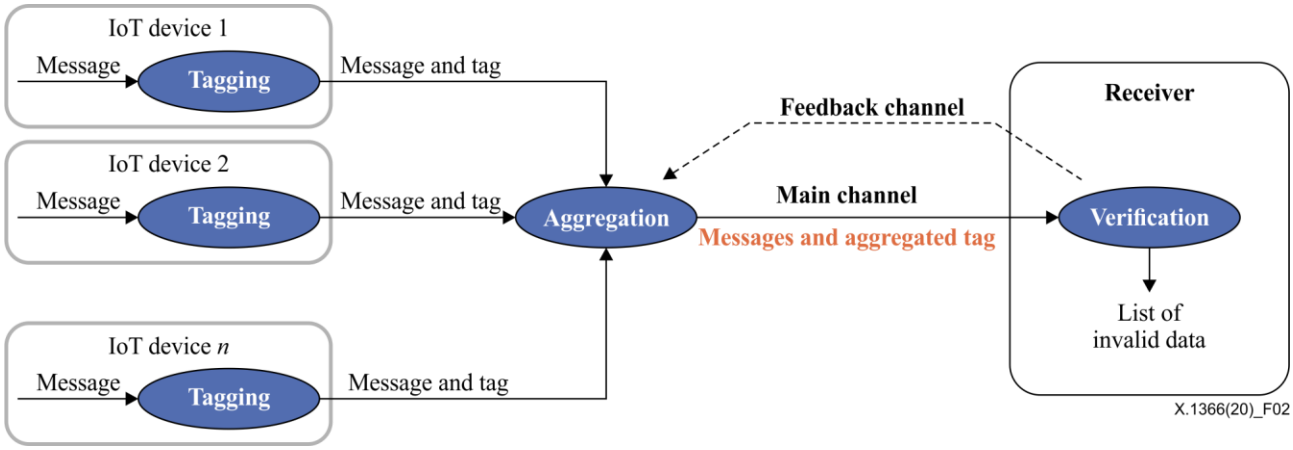

Figure 2 – Basic concept of aggregate message authentication system

NOTE – In a situation where multiple devices send privacy data by using Encrypt-then-MAC schemes, the aggregation technique in this Recommendation can be applied to compress multiple MAC-tags.

In this Recommendation, there are four processes that are used to perform AMA and IAMA schemes: key generation, tagging, aggregation and verification as follows:

1) Key generation takes as input a security parameter and an *ID*, and produces a secret key for the *ID*.

2) Tagging takes a message, an *ID*, and a secret key corresponding to the *ID* as input, and outputs a tag.

3) Aggregation takes multiple tuples of IDs, messages, and tags from multiple devices as input, and produces a tuple of aggregate tags as output.

4) Verification takes all secret keys, multiple pairs of IDs and messages from multiple devices, and a tuple of aggregate tags as input. It specifies invalid messages, and outputs a list of IDs of devices whose messages are invalid.

7 Aggregate message authentication

7.1 General

The AMAC scheme outlined in this Recommendation provides the functionalities of both aggregating multiple MAC-tags into a shorter tag and identifying invalid messages from it. This clause explains how the four algorithms: key generation, tagging, aggregation and verification are constructed to generate an AMAC.

7.2 Specific notation

In this Recommendation, the following specific notations are used:

n: Number of devices.

d: Number of invalid messages from devices.

id: *ID* of a device. Let $ID = \{id_1, id_2, \dots, id_n\}$ be the set of all IDs.

m: Message.

k_{id}: Secret key of a device *id.* For simplicity, k_i denotes the secret key corresponding to id_i instead of k_{id_i}.

$F()$: MAC function which takes a secret key and a message as input and outputs a MAC tag.

$G = (g_{i,j})$: d-disjunct matrix with u rows and n columns. The matrix G has entries in $\{0,1\}$, and columns are indexed by IDs, $id_1, id_2, \dots, id_n$. G is said to be d-disjunct, if the Boolean sums of any d columns of G does not contain any other column, where $x = (x_1, x_2, \dots, x_u)$ *contains* $y = (y_1, y_2, \dots, y_u)$ if $x_i \geq y_i$ for every $1 \leq i \leq u$.

$I(G, i)$: The set of j $(1 \leq j \leq n)$ such that $g_{i,j} = 1$ for every $i = 1,2, \dots, u$.

$\oplus$: Bitwise XOR (exclusive OR) operation.

$H()$: Hash function.

7.3 Algorithm specification

To cover wider applications, two kinds of aggregate MACs with detecting functionality are provided. One is XOR-based (clause 7.3.1) and the other is based on a hash function (clause 7.3.2).

7.3.1 XOR-based construction

7.3.1.1 Key generation

For each id , this process generates a random key. It is denoted by k_{id}.

7.3.1.2 Tagging

Tagging takes a message, an *ID*, and a secret key corresponding to the *ID*, denoted by id, m, k_{id}, respectively, as input, and outputs a MAC tag *t* which is computed by $F(k_{id}, m)$.

7.3.1.3 Aggregation

Aggregation takes IDs, messages, and their MAC tags from *n* devices as input which is denoted by $(id_1, m_1, t_1), \dots, (id_n, m_n, t_n)$. For each i $(1 \leq i \leq u)$, take bitwise XOR of MAC tags whose

corresponding IDs are included in $I(G,i)$ and define it as T_i, namely $T_i = \oplus_{j \in I(G,i)} t_j$. Then, output $(T_1, T_2, \ldots, T_u)$ as an aggregate tag.

7.3.1.4 Verification

Verification takes all the secret keys denoted by $(k_1, \ldots, k_n)$, multiple pairs of IDs and messages from n devices denoted by $(id_1, m_1), \ldots, (id_n, m_n)$, and an aggregate tag denoted by $(T_1, T_2, \ldots, T_u)$ as input. Then, it outputs a list J after the following procedure.

Step 1: $J \leftarrow \{id_1, id_2, \ldots, id_n\}$

Step 2: For $i = 1,2, \ldots, u$, do the following:

If $T_i = \oplus_{j \in I(G,i)} t_j$, then $J \leftarrow J \setminus \{id_j\}$ for all $j \in I(G,i)$.

7.3.2 Hash-based construction

7.3.2.1 Key generation

For each id , this process generates a random key. It is denoted by k_{id}.

7.3.2.2 Tagging

Tagging takes a message, an *ID*, and a secret key corresponding to the *ID*, denoted by id, m, k_{id}, respectively, as input, and outputs a MAC tag *t* which is computed by $F(k_{id}, m)$.

7.3.2.3 Aggregation

Aggregation takes IDs, messages, and their MAC tags from *n* devices as input which is denoted by $(id_1, m_1, t_1), \ldots, (id_n, m_n, t_n)$. For each i $(1 \leq i \leq u)$, compute a hash value of MAC tags whose corresponding IDs are included in $I(G,i)$ and define it as T_i, namely $T_i = H(t_{j_1}, t_{j_2}, \ldots)$ where $I(G,i) = \{j_1, j_2, \ldots\}$ with $1 \leq j_1 < j_2 < \cdots$.

Then, output $(T_1, T_2, \ldots, T_u)$ as an aggregate tag.

7.3.2.4 Verification

Verification takes all the secret keys denoted by $(k_1, \ldots, k_n)$, multiple pairs of IDs and messages from *n* devices denoted by $(id_1, m_1), \ldots, (id_n, m_n)$, and an aggregate tag denoted by $(T_1, T_2, \ldots, T_u)$ as input. Then, it outputs a list J after the following procedure.

Step 1: $J \leftarrow \{id_1, id_2, \ldots, id_n\}$

Step 2: For $i = 1,2, \ldots, u$, do the following:

If $T_i = H(t_{j_1}, t_{j_2}, \ldots)$ where $I(G,i) = \{j_1, j_2, \ldots\}$ with $1 \leq j_1 < j_2 < \cdots$,

then $J \leftarrow J \setminus \{id_j\}$ for all $j \in I(G,i)$.

8 Interactive aggregate message authentication

8.1 General

The IAMA scheme proposed in this Recommendation provides the functionality so that IAMA can identify invalid messages with a smaller amount of tag size than those of the AMA scheme in clause 7. An IAMA scheme consists of two algorithms, key generation and tagging, and an interactive protocol between aggregation and verification. This clause explains how those algorithms and the protocol are constructed.

8.2 Specific notation

In this Recommendation, the following specific notations are used:

n: Number of devices.

d: Number of invalid messages from devices.

id: *ID* of a device. Let $ID = \{id_1, id_2, \dots, id_n\}$ be the set of all IDs.

m: Message.

k_{id} : Secret key of a device *id*. For simplicity, k_i denotes the secret key corresponding to id_i instead of k_{id_i}.

$F()$: MAC function which takes a secret key and a message as input and outputs a MAC tag.

AGT: Adaptive group testing protocol.

$\oplus$: Bitwise XOR (exclusive OR) operation.

$H()$: Hash function.

8.3 Specification of interactive protocol

An IAMA can be constructed from a MAC function *F*() and an AGT, see Appendix III for adaptive group testing. Such constructions are presented here by using two kinds of operations, XOR or a hash function as presented in the constructions of AMA.

8.3.1 XOR-based construction

8.3.1.1 Key generation

For each id , this process generates a random key. It is denoted by k_{id}.

8.3.1.2 Tagging

Tagging takes a message, an *ID*, and a secret key corresponding to the *ID*, denoted by id, m, k_{id}, respectively, as input, and outputs a MAC tag *t* which is computed by $F(k_{id}, m)$.

8.3.1.3 Aggregation and verification

Aggregation and verification are constructed based on an AGT protocol as shown in Figure 3. Aggregation takes the whole set of IDs $ID = \{id_1, id_2, \dots, id_n\}$, messages, and their MAC tags from *n* devices as input which is denoted by $(m_1, t_1), \dots, (m_n, t_n)$, where (m_i, t_i) $(1 \leq i \leq n)$ is a message-tag pair corresponding to id_i. Verification takes the whole set of IDs ID and all secret keys k_i $(1 \leq i \leq n)$ corresponding to id_i. First, aggregation selects a subset $S \subseteq ID$, generates an aggregated tag T_S by compressing MAC tags of S: T_S can be generated by taking XOR of MAC tags, $T_S = \oplus_{j \in S} t_j$. Then, Aggregation sends T_S with messages $(m_1, \dots, m_n)$ to verification. Next, verification sets $J = ID$, and checks validity of T_S by using secret keys of S: T_S is regarded as valid if $T_S = \oplus_{j \in S} t_j$, where $t_j = F(k_j, m_j)$; otherwise, T_S is regarded as invalid. If T_S is valid, set $J \leftarrow J \setminus S$. Verification sends the checking result of T_S (i.e., one-bit information) to aggregation. Then, aggregation selects another subset $S' \subseteq ID$, generates an aggregated tag $T_{S'}$ by compressing MAC tags of S', and sends $T_{S'}$ to verification. Verification checks validity of $T_{S'}$ by using secret keys of S'; If $T_{S'}$ is valid, $J \leftarrow J \setminus S'$. Verification sends the checking result of $T_{S'}$ to aggregation. After repeating the above procedures between aggregation and verification, verification finally outputs a list J which consists of IDs of devices whose messages are invalid.

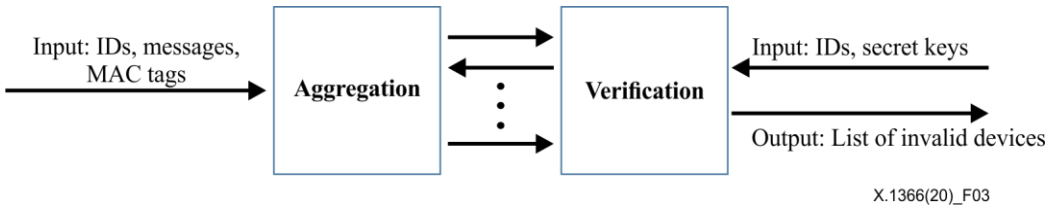

Figure 3 – Interactive protocol between aggregation and verification

8.3.2 Hash-based construction

8.3.2.1 Key generation

For each id , this process generates a random key. It is denoted by k_{id}.

8.3.2.2 Tagging

Tagging takes a message, an ID, and a secret key corresponding to the ID, denoted by id, m, k_{id}, respectively, as input, and outputs a MAC tag *t* which is computed by $F(k_{id}, m)$.

8.3.2.3 Aggregation and verification

Aggregation and verification are constructed based on an AGT protocol as shown in Figure 3. Aggregation takes the whole set of IDs $ID = \{id_1, id_2, \dots, id_n\}$, messages, and their MAC tags from *n* devices as input which is denoted by $(m_1, t_1), \dots, (m_n, t_n)$, where (m_i, t_i) $(1 \leq i \leq n)$ is a message-tag pair corresponding to id_i. Verification takes the whole set of IDs ID and all secret keys k_i $(1 \leq i \leq n)$ corresponding to id_i. First, aggregation selects a subset $S \subseteq ID$, generates an aggregated tag T_S by computing a hash value $T_S = H(t_{j_1}, t_{j_2}, \dots)$ where $S = \{id_{j_1}, id_{j_2}, \dots\}$ with $1 \leq j_1 < j_2 < \cdots$. Then, aggregation sends T_S with messages $(m_1, \dots, m_n)$ to verification. Next, verification sets $J = ID$, and checks validity of T_S by using secret keys of S: T_S is regarded as valid if $T_S = H(t_{j_1}, t_{j_2}, \dots)$, where $t_j = F(k_j, m_j)$; otherwise, T_S is regarded as invalid. If T_S is valid, set $J \leftarrow J \setminus S$. Verification sends the checking result of T_S (i.e., one-bit information) to aggregation. Then, aggregation selects another subset $S' \subseteq ID$, generates an aggregated tag $T_{S'}$ by compressing MAC tags of S', and sends $T_{S'}$ to verification. Verification checks validity of $T_{S'}$ by using secret keys of S'; If $T_{S'}$ is valid, $J \leftarrow J \setminus S'$. Verification sends the checking result of $T_{S'}$ to aggregation. After repeating the above procedures between aggregation and verification, verification finally outputs a list J which consists of IDs of devices whose messages are invalid.

Annex A

Guidance and limitations

(This annex forms an integral part of this Recommendation.)

A.1 Guidance on use of aggregate message authentication (AMA)

This Recommendation discusses the applicability of embedding an aggregate node into existing message authentication code (MAC) protocols without changing the input-formats or network connections of underlying MACs. In addition, aggregation is a keyless procedure, and does not need maintenance of any secret key in the aggregation node. Furthermore, the aggregation is executed only by computing bitwise XOR operations or hash functions, and hence, the AMA schemes in this Recommendation are suitable to use for authentication in a lightweight manner.

In the aggregation processing described in clause 7, a *d*-disjunct matrix needs to be generated and stored. Several methods such as those described in [b-TM05] to generate *d*-disjunct matrices are known, and it is also possible to use a compressed form of a *d*-disjunct matrix as described in [b-MK19]. This Recommendation proposes utilizing those techniques even in AMA schemes. For a d-disjunct matrix with u rows and n columns, the AMA scheme is more effective than the traditional one-to-one authentication if $u < n$; and more effective if $d \ll \sqrt{n}$.

NOTE − Security levels of AMA (or IAMA) schemes constructed by bitwise XOR or hash functions are described here according to schemes described in [b-HS18] and [b-SS19]. There are three kinds of security notions, unforgeability, identifiability-completeness, and identifiability-(weak)-soundness: Unforgeability guarantees that no message can be forged; Identifiability-completeness guarantees that any valid message is judged as valid by the scheme; Identifiability-soundness guarantees that any invalid message is judged as invalid by the scheme, while identifiability-weak-soundness is the same as identifiability-soundness except that an adversary is supposed to obtain no valid MAC-tags and to corrupt no devices before attacking. The weak-soundness is still useful in applications, since it covers message tampering.

Security levels of AMA (or IAMA) schemes in this Recommendation are described as follows. XOR-based construction meets unforgeability, identifiability-completeness, and identifiability-weak-soundness if the underlying MAC meets unforgeability. Hash-based construction meets unforgeability, identifiability-completeness, and identifiability-soundness, if the underlying MAC meets unforgeability and the hash function is regarded as a random function.

A.2 Limitations of the use of AMA

This Recommendation assumes that the number of invalid messages is at most *d* in AMA schemes, and this parameter is set as a system parameter. This means that there is a need to estimate the number *d* beforehand.

What happens if the number of invalid messages exceeds the assumed value *d*? In this case, verification finally outputs a list *J* that contains more than *d* IDs of devices; the IDs of devices that had sent invalid messages are included in the list *J*; but, some IDs of devices that had not sent invalid messages may also be included in the list *J*. In this case, it is recommended to set up a larger value *d* for the AMA scheme again.

Annex B

Combination with existing one-to-one authentication protocols

(This annex forms an integral part of this Recommendation.)

AMA (or IAMA) schemes in this Recommendation can be utilized in combination with the traditional one-to-one authentication. The traditional one-to-one authentication is understood as the AMA scheme where the underlying disjunct matrix is the identity matrix. For $n = n_1 + n_2$ devices, if it is preferable to aggregate only n_1 MAC-tags among n MAC-tags, do the following: apply an AMA (or IAMA) scheme for the n_1 devices, and apply the one-to-one authentication for the other n_2 devices, as shown in Figure B.1.

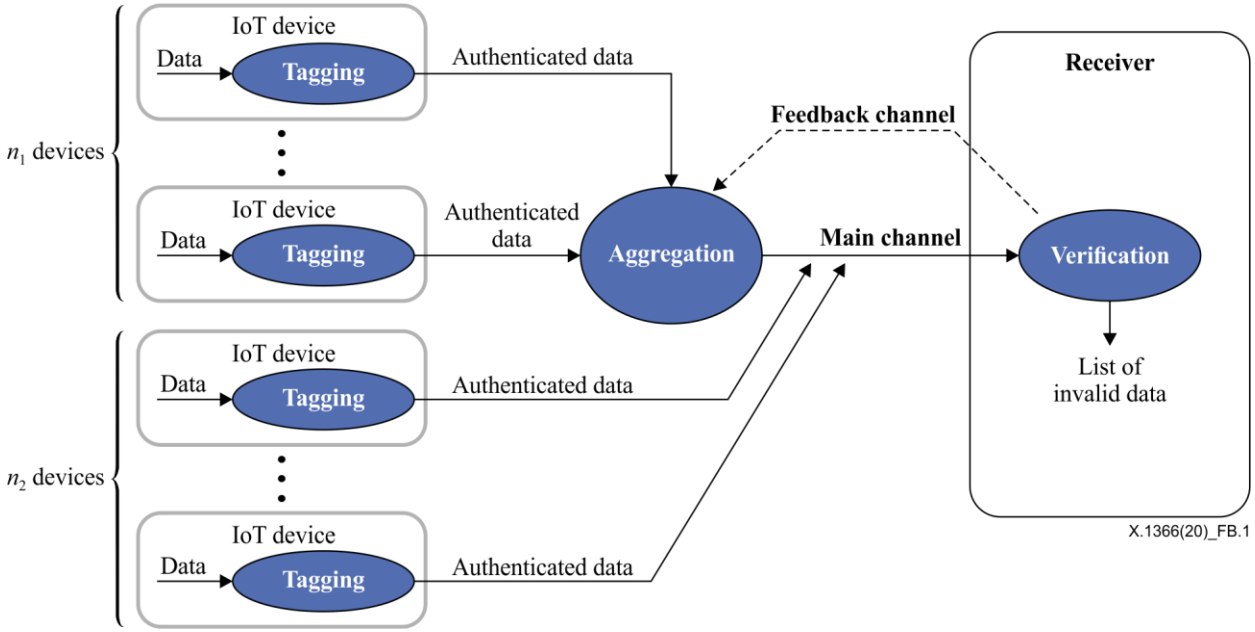

Figure B.1 – Combination with one-to-one authentication protocols

Appendix I

Use cases on the use of AMA

(This appendix does not form an integral part of this Recommendation.)

I.1 Introduction

The aggregate message authentication scheme can be applied for ensuring entity (identity) authentication as well as for ensuring message authentication. Furthermore, the scheme may not be applicable in all use cases for utilizing Internet of things (IoT) devices. Specifically, this scheme is quite effective and suitable for use cases in the following conditions:

- Message authentication is required from tens to tens of thousands of IoT devices.
- The data/message being handled for the authentication process occurs frequently and intermittently.

The following are examples of applications that it can be specifically assumed would utilize the aggregate authentication technology:

a) Applications for concisely and frequently sending data/messages such as semi-movie (still image) data
   - Surveillance applications using image data

b) Applications for remote telemetry:
   - Applications for monitoring of factory operations
   - Audience dynamics surveying applications
   - Health monitoring applications such as Citizen Marathon for example
   - Applications for management of facilities such as streetlights installed in urban areas
   - Applications for traffic monitoring
   - Applications for river level monitoring

By applying this aggregate authentication technology in the above IoT applications, the efficiency of message transmission and authentication processing in the entire IoT system can be dramatically improved.

The following use cases are examples for utilizing this aggregate authentication scheme specified in this Recommendation.

I.2 Use case-1: Theme parks and leisure centres

In the case of parks and leisure centres, etc., it can be assumed that there are between 1 000 to 10 000 visitors at the same time. That is, thousands of visitors who have appropriate privileges for use of attractions in the park/centre that may need to be verified at the same time. In this case, an aggregated authentication scheme may be perfectly suitable to perform efficient authorization management. As shown in Figure I.1, aggregate servers can be in each attraction facility to collect and aggregate authentication tags for requesting verification from the back-end authentication server.

More specifically, visitors purchase in advance an admission ticket in which information on events, attractions admission, and web services to be provided are embedded in chips. This technology is widely used in marathons. Wristbands with embedded chips can be also considered as an alternative to admission tickets.

At the event venue main gate or at individual gates for each attraction, the contents of the entrance ticket are read, aggregated using aggregate authentication technology, and sent to the aggregate authentication server.

The aggregate authentication server centre analyses the service contents and requirements provided to visitors, informs various attraction venues and web service providers, and uses it for analysis and prediction of congestion. After verification, visitors can utilize various registered web services using their own smartphones or glass-type wearable devices for example.

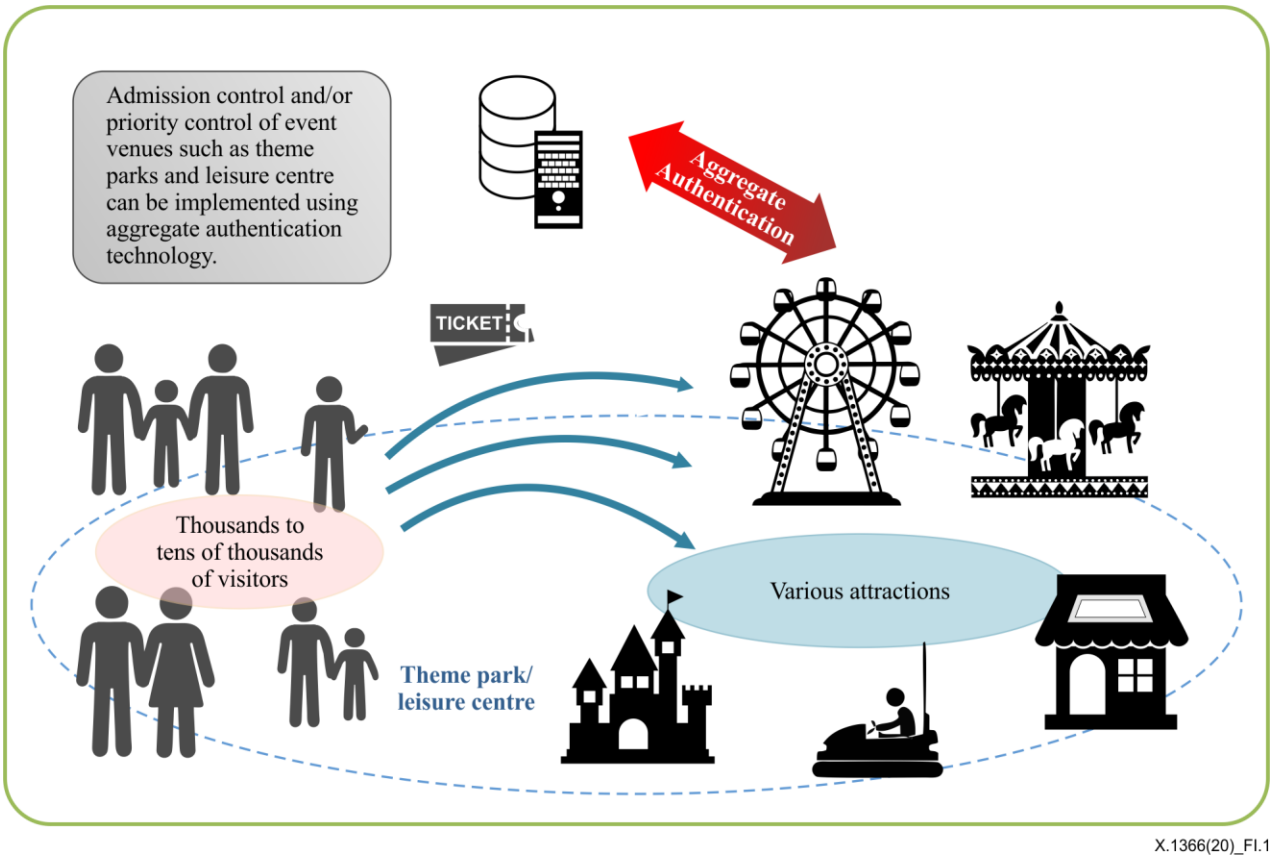

Figure I.1 – Aggregate authentication scheme in theme parks and leisure centres

I.3 Use case-2: Surveillance sensors

I.3.1 General

In order to be alerted to and intervene at an early stage in natural disasters and accidents/incidents, monitoring of activities using surveillance sensors such as cameras for IoT devices can be a use case of the aggregate authentication schemes provided by this Recommendation. In this case, quasi-video or still images captured by a number of surveillance cameras are sent to a surveillance centre in almost real time (or periodically) but it is important to ensure the reliability and integrity of the data being sent.

However, when the number of surveillance sensors becomes very high, it is no longer efficient to verify an authentication code with the image data from each camera by checking the authentication code of each camera one by one. In such an environment, the aggregate authentication scheme is effective. Authentication codes with data can be aggregated in the aggregate servers before being sent to the surveillance centre, so that the entire IoT system can provide efficient authentication and communications. The number of aggregation servers is dependent on the number of surveillance sensors. Figure I.2 shows surveillance sensors in an aggregate authentication scheme.

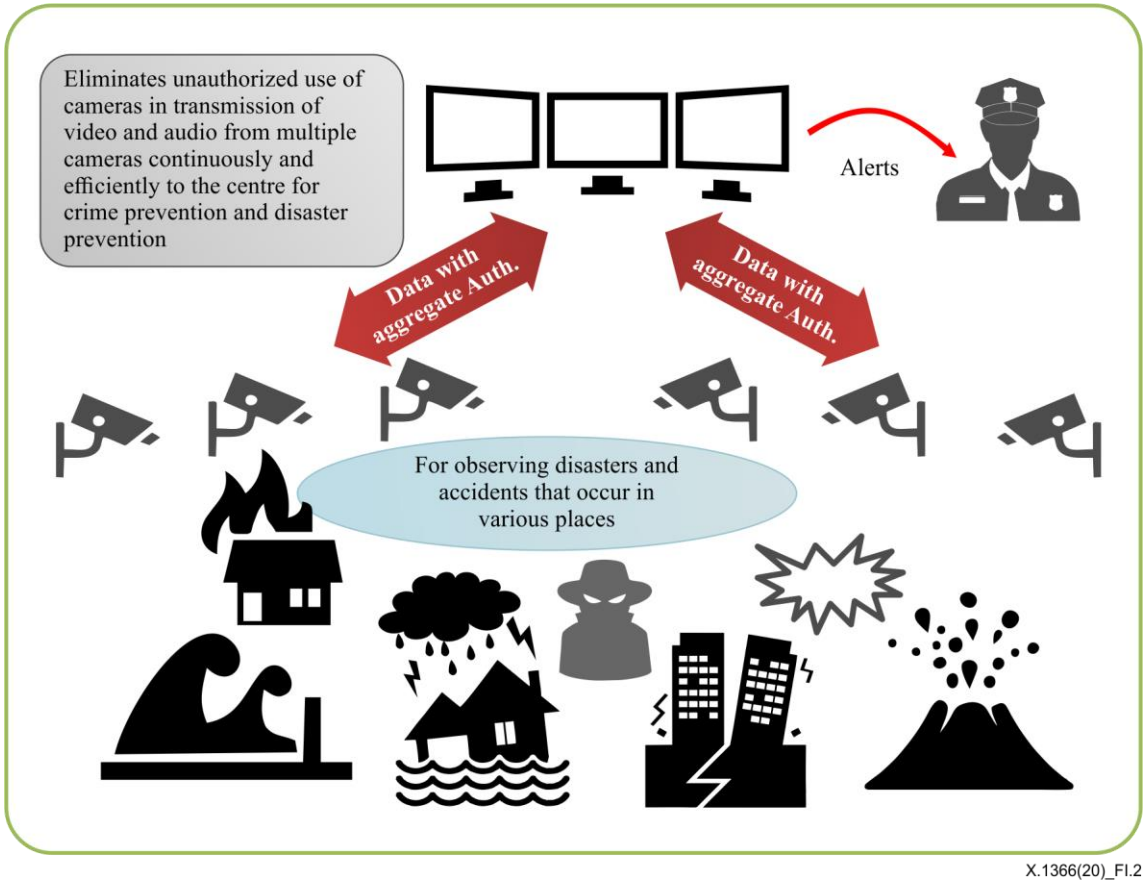

Figure I.2 – Surveillance sensors in an aggregate authentication scheme

I.3.2 Specific use cases

1) Monitoring of living environments such as communities and housing

Living environment information from various sensors such as surveillance cameras attached to apartment block buildings, smart communities, private homes, etc., is aggregated at a gateway (IoT hub) and transmitted to a centre server using aggregation authentication technology.

The centre will analyse the information received, and use it to help monitoring the living environment, predicting abnormalities and failures, responding promptly, and preventing crime and disasters.

More specifically, data captured in various environmental sensors, home appliance sensors, surveillance cameras, door/window opening/closing status sensors, gas/water/electricity infrastructure operation status sensors, elevator monitoring sensors, etc., are sent to external centres. Aggregation authentication schemes using both terminal authentication and data authentication are effective as an authentication means for collecting diverse and large amounts of data and transmitting the data efficiently.

2) Maintenance and monitoring of social infrastructure, disaster response

The maintenance and management of social infrastructure such as bridges, tunnels, and roads using IoT is being introduced in various fields, and it is widely expected that IoT services will play an extremely important role in achieving safe and secure society in the near future. For example, in the cases of aging bridges and elevated roads, relevant data such as strains, vibrations, displacement, inclination, etc., and video information is captured in detail by various sensors. The volume of data which should be sent to the centre is becoming extremely large.

Currently the aggregate authentication method is proving very effective as one of the measures used to improve the efficiency of wireless Internet circuit use and to avoid congestion. In addition to the maintenance and management of these social infrastructures, it is also possible to apply the aggregate

authentication method to the gateway of the IoT system for use of systems for the constant monitoring of water levels and flow changes in rivers and lakes in agricultural environments.

3) Disaster prevention systems using surveillance cameras

Surveillance cameras are installed and operated for various purposes including crime prevention and disaster prevention at various places all over the world. Generally, in a network that handles image and audio information, it is necessary to continuously transmit a large amount of data to the centre side, and it is effective to apply an aggregation authentication technique for efficient transmission. That is, it is possible to improve communication efficiency between the IoT device and the IoT gateway and between the IoT gateway and the centre by applying the aggregate authentication method.

4) Logistics monitoring, improving the efficiency of transportation systems

In logistics and transportation business systems, IoT systems are increasingly utilized for improving the efficiency and high functionality of the business. For example, a solution that precisely manages status information of goods and packages from shipping to delivery is being put to practical use in various fields. In such a system, more stable and efficient logistics management can be achieved by applying aggregate authentication technology to the system that sends various sensor information on all packages to the center. It is also conceivable to provide an IoT gateway for vehicles such as cars equipped with an enormous number of sensors and apply aggregate authentication technology at the vehicle gateway for transportation systems.

Appendix II

Related activities on AMA schemes

(This appendix does not form an integral part of this Recommendation.)

An AMA scheme, that differs from the scheme described in clause 7, was first proposed by Katz and Lindell in [b-KL08] and allows aggregation of multiple MAC-tags of multiple messages into a shorter tag. Specifically, Katz and Lindell [b-KL08] formalized the model and security of AMA and provided the simple construction of AMA by taking bitwise XOR of all MAC tags. It is possible to verify the validity of multiple messages with only a shorter single tag, however, it is generally impossible to identify invalid messages in their AMA scheme once multiple messages are judged invalid with respect to the single tag. The AMA schemes in this Recommendation achieve both the functionalities of aggregating multiple MAC-tags into a shorter tag and identifying invalid messages from it. The AMA code provided in clause 7 of this Recommendation is based [b-HS18], while the interactive authentication protocol for use of AMA provided in clause 8 is based on [b-SS19].

Appendix III

Adaptive group testing protocol

(This appendix does not form an integral part of this Recommendation.)

Group testing, as discussed in [b-DH00], is a method for specifying special items called defectives among a large number of whole items using a small number of tests rather than carrying out individual testing for each item.

In the following example of group testing protocol shown in Figure III.1, it is supposed that there are total of *n* items of which there are *d* defectives.

In adaptive group testing, tests can be carried out several times such that a subset of items to be tested can be selected after observing the results of the previous test. A competitive group testing is an adaptive group testing which does not need to know the number *d* of defectives beforehand.

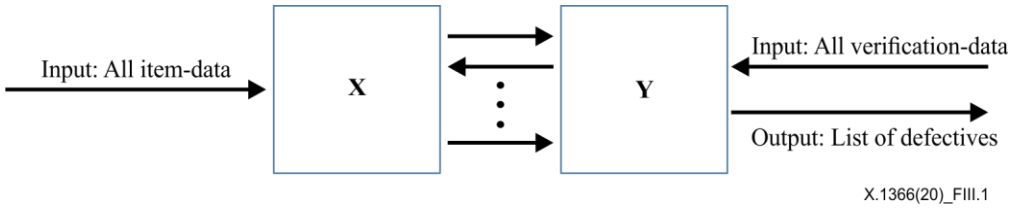

Figure III.1 – Adaptive group testing protocol

Formally, adaptive group testing is an interactive protocol between X and Y as shown in Figure III.1.

X takes the whole set of IDs $ID = \{id_1, id_2, \ldots, id_n\}$ and all item-data $\mathrm{data_i}$ ($1 \leq i \leq n$) corresponding to id_i. Y takes the whole set of IDs ID and all verification-data $\mathrm{ans_i}$ ($1 \leq i \leq n$) corresponding to id_i. First, X selects a subset $S \subseteq ID$, generates $\mathrm{test_S}$ by compressing item-data of S, and sends $\mathrm{test_S}$ to Y. Next, Y sets $J = ID$, and checks validity of $\mathrm{test_S}$ by using verification-data of S. If $\mathrm{test_S}$ is valid, set $J \leftarrow J \setminus S$. Y sends the checking result of $\mathrm{test_S}$ (i.e., one-bit information) to X. Then, X selects another subset of ID, and repeat the procedures between X and Y. After repeating the above procedures between X and Y, Y finally outputs a list J which consists of IDs of defectives.

For instance, adaptive group testing protocols include the binary search, the rake-and-winnow algorithm [b-EGH07], Li's multi-stage algorithm [b-Li62], and the digging algorithm described in clause 4.6 of [b-DH00].

Bibliography

[b-ITU-T X.813] Recommendation ITU-T X.813 (1996), *Information technology – Open Systems Interconnection – Security frameworks for open systems: Non-repudiation framework*.

[b-DH00] D. Z. Du and F. K. Hwang, *Combinatorial Group Testing and Its Applications*, Series on Applied Mathematics, vol. 12, 2nd edn. World Scientific, Singapore, 2000.

[b-EGH07] D. Eppstein, M. T. Goodrich, and D. S. Hirschberg, *Improved Combinatorial Group Testing Algorithms for Real-world Problem Sizes*, SIAM J. Comput. 36(5), pp. 1360-1375, 2007.

[b-HS18] S. Hirose and J. Shikata, *Non-adaptive Group-Testing Aggregate MAC Schemes*, The 14th International Conference on Information Security Practice and Experience (ISPEC 2018), LNCS 11125, pp. 357-372, Springer, 2018.

[b-KL08] J. Katz and A.Y. Lindell, *Aggregate message authentication codes*, CT-RSA 2008, LNCS 4964, pp. 155-169. Springer, 2008.

[b-Li62] C. H. Li, *A Sequential Method for Screening Experimental Variables*, J. Am. Stat. Assoc. 57 (298), pp. 455-477, 1962.

[b-MK19] K. Minematsu and N. Kamiya, *Symmetric-key Corruption Detection: When XOR-MACs meet combinatorial group testing*, ESORICS 2019, Part I, LNCS 11735, pp. 595-615, Springer, 2019.

[b-MOV96] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, October 1996, Fifth Printing (August 2001).

[b-SS19] S. Sato and J. Shikata, *Interactive Aggregate Message Authentication Scheme with Detecting Functionality*, The 33rd International Conference on Advanced Information Networking and Applications (AINA-2019), pp. 1316-1328, Springer, 2019.

[b-TM05] N. Thierry-Mieg, *A New Pooling Strategy for High-throughput Screening: the Shifted Transversal Design*, *BMC Bioinformatics*, vol. 7, no. 28, 2005.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	**Data networks, open system communications and security**
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2020