INTERNATIONAL STANDARD

ISO/IEC 27551

First edition
2021-09

Information security, cybersecurity and privacy protection — Requirements for attribute-based unlinkable entity authentication

Sécurité de l'information, cybersécurité et protection de la vie privée — Exigences relatives à l'authentification des entités non rattachables par des attributs

Reference number
ISO/IEC 27551:2021(E)

COPYRIGHT PROTECTED DOCUMENT

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org
Published in Switzerland

Contents

Page

Foreword ..... **iv**
Introduction ..... **v**
1 Scope **1**
2 Normative references **1**
3 Terms and definitions **1**
4 Symbols and abbreviated terms **2**
5 General objectives of attribute-based entity authentication **2**
6 Properties of attribute-based entity authentication protocols **4**
6.1 Correctness 4
6.2 Unforgeability 4
6.2.1 General 4
6.2.2 Replay protections 4
7 Unlinkability properties of attribute-based entity authentication protocols **4**
7.1 General 4
7.2 Generic definition of unlinkability 5
7.3 Specific definitions of unlinkability 5
7.3.1 General 5
7.3.2 Passive outsider unlinkability (anti-tracking from passive outsiders) 7
7.3.3 Active outsider unlinkability (anti-tracking from active outsiders) 7
7.3.4 RP-U unlinkability ("anonymous visits" to an RP) 7
7.3.5 AP-U unlinkability 8
7.3.6 RP+AP-U unlinkability (anti-RP-AP-collusion) 8
7.3.7 AP-RP unlinkability (anti-tracking of RP from AP) 8
7.3.8 AP-RP+U unlinkability 8
7.3.9 RP+RP'-U unlinkability (anti-tracking of U from a set of colluding RPs) 8
7.4 Relationships between notions of unlinkability 9
7.5 Unlinkability levels for attribute-based entity authentication 9
7.6 Models 10
8 Attributes **10**
8.1 Categories of attributes 10
8.1.1 Personal attributes 10
8.1.2 Self-claimed attributes 10
8.1.3 Verified attributes 10
8.1.4 Static attributes 11
8.1.5 Semi-static attributes 11
8.1.6 Dynamic attributes 11
8.1.7 Computed attributes 11
8.1.8 Identifying attributes 11
8.1.9 Supporting attributes 11
8.2 Verified attribute expiry and revocation 11
8.3 Attribute assurance 11
9 Requirements for level N attribute-based unlinkable entity authentication **11**
Annex A (informative) **Formal definitions for security and unlinkability notions** **13**
Annex B (informative) **Examples of attribute-based entity authentication protocols** **19**
Annex C (informative) **26**
Annex D (informative) **Use cases for attribute-based unlinkable entity authentication** **33**
Bibliography **34**

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see ~~www.iso .org/directives~~ or ~~www.iec .ch/members _experts/refdocs~~).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see ~~www.iso.org/patents~~) or the IEC list of patent declarations received (see ~~patents.iec.ch~~).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see ~~www.iso .org/iso/foreword .html~~. In the IEC, see ~~www.iec .ch/understanding-standards~~.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 27, *Information security, cybersecurity and privacy protection*.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at ~~www.iso .org/members .html~~ and ~~www.iec .ch/national -committees~~.

Introduction

ISO/IEC 29100 sets forth eleven privacy principles which apply to all actors that can be involved in the processing of PII. The second principle is the collection limitation. Despite this recommendation, the current state of the art is that internet sites collect more than necessary information during the PII principal's access to the service. For example, if the site only requires verification that the PII principal is over a certain age, only that information should be necessary for the consumption of the service. However, it is often the case that other information such as the user's persistent identifier is supplied, making it possible to link visits from the same PII principal to different sites or to link two or more visits from the same PII principal to the same site.

To adhere to the principle of the collection limitation, the site in the above case should instead use a type of entity identifier that does not allow the site to link two or more visits by the PII principal. This means that, when two transactions are performed, it is difficult to distinguish whether the transactions were performed by the same user or by two different users. This is one type of unlinkability. Several other types of unlinkability can also be considered and desired in applications.

Attribute-based unlinkable entity authentication (ABUEA) provides a means for PII principals to establish the authenticity of a selected subset of their identity attributes without revealing a larger subset. Special focus is put on unlinkability and a metric that measures the strength of this property in implementations of ABUEA is introduced. This document focuses on cases where at least one attribute is attested by a third party. This document also identifies security properties to be met to achieve various protections as well as unlinkable properties.

The methodology developed by this document may be tailored and applied to other privacy principles. The requirements identified in this document apply at the application communication layer. However, some properties met at the application layer protocol can be ruined by a lower layer protocol, such as the network layer, which means that the lower layers' privacy and security properties should also be taken into consideration to ensure that the properties met at the application communication layer are still valid when considering the privacy and security characteristics of the lower communication layers.

Information security, cybersecurity and privacy protection — Requirements for attribute-based unlinkable entity authentication

1 Scope

This document provides a framework and establishes requirements for attribute-based unlinkable entity authentication (ABUEA).

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 24760-1, *IT Security and Privacy — A framework for identity management — Part 1: Terminology and concepts*

ISO/IEC 29100, *Information technology — Security techniques — Privacy framework*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 29100, ISO/IEC 24760-1, and the following apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at ~~https://www.iso .org/obp~~

— IEC Electropedia: available at ~~https://www.electropedia .org/~~

3.1
anonymity set
set of identities that shares certain characteristics

3.2
attribute provider
authority trusted by one or more users and one or more relying parties to issue or verify attributes related to an entity

3.3
significantly
not vanishing faster than any inverse polynomial in the security parameter

3.4
user-agent
software and/or hardware used by the PII Principal to interact with the system

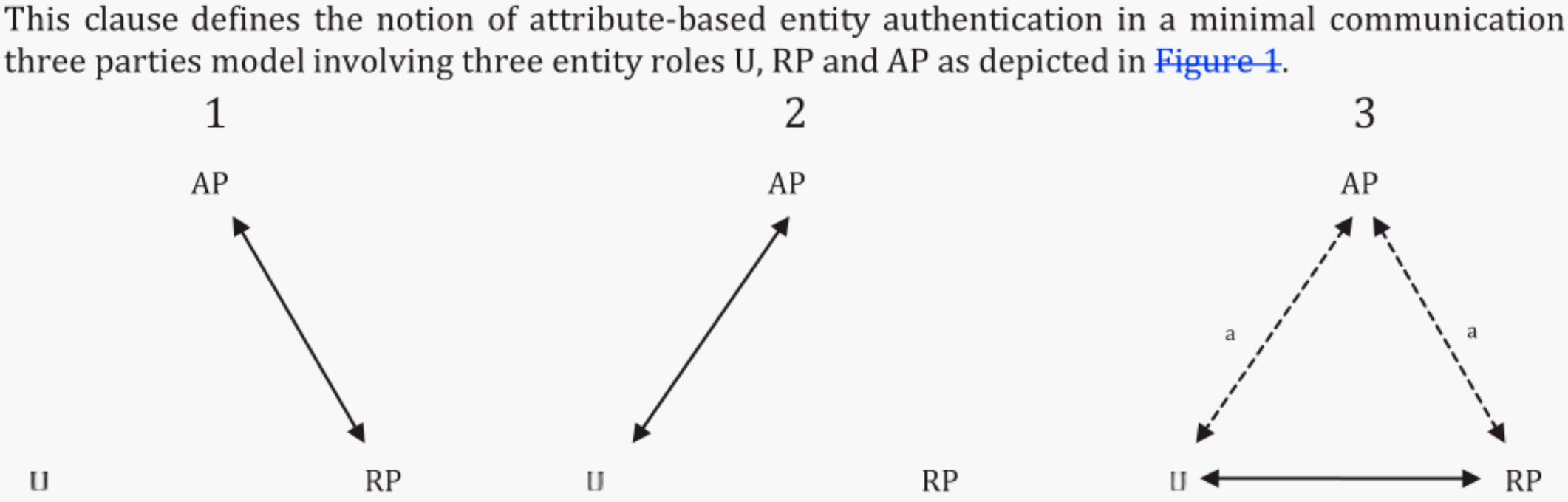
4 Symbols and abbreviated terms

A	adversary
AO	active outsider
AP	attribute provider
OIDC	OpenID Connect
PII	personally identifiable information
PO	passive outsider
RP	relying party
SIOP	self-issued OpenID provider
U	user-agent
UL	unlinkability level

5 General objectives of attribute-based entity authentication

Attribute-based entity authentication is a means to establish a form of trust between two unfamiliar parties that share trust in a common third-party entity.

This clause defines the notion of attribute-based entity authentication in a minimal communication three parties model involving three entity roles U, RP and AP as depicted in ~~Figure 1~~.



Key

1 setup phase (optional)
2 user registration phase
3 authentication phase
[a] Optional.

Figure 1 — Phases of attribute-based entity authentication

RP trusts AP in the sense that RP is convinced of the correctness of statements expressed by AP. RP and AP can have engaged in some optional preliminary procedure, referred to as the setup phase, which enables the RP to ascertain that statements expressed by AP are genuine.

A PII principal, referred to as a user hereafter, uses a software called user-agent or U to communicate with AP and RP. U has a number of attributes which collectively represent its distinct identity. U and AP can also have carried out a preliminary procedure referred to as the user registration phase, in which AP validates the user's attributes and links them to U's attributes. During this process, U can be given data material to enable later attribute-based entity authentication towards RP.

There is no such preliminary procedure between U and RP, meaning that U and RP are a priori strangers to one another.

An attribute-based entity authentication protocol is a sequence of computations and communications among U, RP and AP which, when conducted successfully throughout, results in a state at RP where RP is convinced that a statement made by U about its attributes is correct or not. The purpose of the protocol is to reach that state.

The authentication phase is the protocol stage where U and RP interact, which can involve the participation of AP or not.

The description of a particular attribute-based entity authentication protocol requires a specification of the attributes, of the statements that can be made on them, as well as of all computations and communications between the three parties. It includes a description of the authentication phase, the setup phase if any, and the user registration phase if any.

NOTE Attribute-based entity authentication can also be achieved in communication models that extend beyond the minimal U-RP-AP model either by involving additional specific-purpose entities or by limiting the use of communication channels at determined stages of the protocol. ~~Annex B~~ describes some examples of attribute-based entity authentication protocols and their underlying model.

Attributes are defined in ISO/IEC 24760-1. As properties, they can have:

— a type, a Boolean, or a character string of alphabetical characters, or an integer in a certain range, or a compound type built on these basic types (such as a fixed length vector of integers or a dynamic list of mixed strings and integers, and so forth);

— a name, which is a string in a prescribed alphabet;

— a value selected within the range of admissible values for the considered type.

Other properties of attributes such as their origin or level of assurance, or more generally classes or categories of sorts, can exist and be involved in the attribute-based authentication protocol. However, they are usually encoded as additional attributes. Therefore, it is enough to rely on the notions of type, name and value when describing an attribute.

A policy decision function is a function that takes a policy and other information for the purpose of returning a boolean value. It is defined as a logic predicate combining basic relational expressions using logic operators such as OR, AND or possibly more complex ones such as threshold gates (t-out-of-n). A relational expression can express:

— equality of an attribute value to a particular value;

— non-equality of an attribute value to a particular value;

— inequality of an attribute value towards a particular value (less than, greater than). This requires that the attribute type support an ordering over its set of admissible values.

It is usual to rely on a structured language to express policies when some level of genericity is desired. OASIS eXtensible Access Control Markup Language (XACML) is one such example. In other applications, the policy may be fixed and hard-coded into the attribute-based entity authentication protocol itself.

It should also be noted that some attribute-based entity authentication protocols may only support restricted policies, where:

— attribute values can only be compared to constants and not to other attribute values;

— the nature or the number of logic operators is limited; or

— some other restriction applies.

The purpose of an attribute-based entity authentication protocol is for RP to be convinced that the set of attributes A_U temporarily associated to an a-priori unknown entity U satisfies a certain policy P,

namely that the policy decision function returns true for P(AU). For attributes that originate from a neutral AP that the RP trusts, some form of interaction with that AP is necessary.

~~Annex D~~ describes examples of use cases in which ABUEA systems are used.

6 Properties of attribute-based entity authentication protocols

6.1 Correctness

Under the assumptions that:

— RP is always convinced by the statements expressed by AP;

— all parties U, AP and RP engage in the correct execution of the protocol;

the protocol is correct when, if the user has a set of attributes AU and AU satisfies the policy P, then the protocol terminates in an acceptance state by the RP, meaning that RP acknowledges that P(AU) = true.

6.2 Unforgeability

6.2.1 General

Unforgeability is a security property that exists for attribute-based entity authentication protocols.

The protocol is unforgeable if it is infeasible for U to make RP terminate the protocol in the acceptance state when P(AU) = false.

~~Clause A.2~~ describes the conditions of achieving unforgeability.

6.2.2 Replay protections

Unforgeability requires two security measures to be taken into consideration for any kind of entity authentication protocol, namely:

— replay protection against one relying party; and

— replay protection against different relying parties.

The first kind of protection requires the use of a time-variant parameter that can either be a challenge sent by the relying party and then reused by the U or be a unique number presented by the U to the SP. These time-variant parameters are part of the computation of the credentials presented by the U to the RP.

The second kind of protection requires the use in the protocol of a data item containing a characteristic unique to the intended relying party.

7 Unlinkability properties of attribute-based entity authentication protocols

7.1 General

In this document, unlinkability refers to a family of properties that an attribute-based entity authentication protocol can or cannot fulfil. The purpose of this clause is to provide a definition for these properties and show how they interrelate.

Note that these definitions are formulated for attribute-based entity authentication protocols operating in the minimal U-RP-AP model. They can give rise to distinct definitions in extended communication models.

7.2 Generic definition of unlinkability

Linking is defined as the ability for an entity or a group of colluding entities to distinguish protocol executions where:

— an entity role is played by the same entity; from

— that entity role is played by different entities.

Unlinkability refers to the inability to link protocol executions. The entity or group of entities attempting to link protocol executions is called the adversary while the entity role under observation is the target entity role. Considering different settings for the adversary and target entity role produces entirely different notions of unlinkability.

For example, if U visits an RP more than twice and if the RP can link these visits together to recognize the repeated visits, then the RP is linking the user visits. In this case, U is not “anonymous” but only “pseudonymous” at best and if U was wishing to be “anonymous”, then RP is acting as an adversary against the user’s wish. Similarly, when U is trying to authenticate itself to RP using attributes provided by AP, U may wish so that AP cannot find out to which RP U has provided those attributes. Under such circumstances, if AP identifies that U provided attributes to RP, then AP is acting as an adversary.

The protocol itself is said to be unlinkable if its executions cannot be linked, given explicit settings for the adversary and target entity role.

NOTE A clear distinction is to be made between the various unlinkability properties that a protocol can or cannot achieve and the traceability or linkability of data transfers at the data transport level. It is usual to assume that an anonymization tool such as TOR[2] can be used to avoid a trivial form of linking through network connections instead of the nature of the exchanged messages. This consideration is independent of the actual unlinkability properties that a protocol possesses or not.

This document is concerned with achieving unlinkability of the protocol executions without external context (metadata). Even if a protocol is unlinkable, linking may be achieved with metadata (e.g. by considering the timings or location when authentication takes place). Techniques to prevent linking via metadata are out of scope for this document.

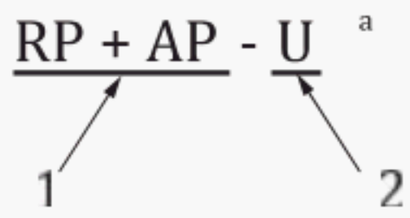
The level of anonymity also depends on the size of the anonymity set resulting from the combination of policy P and the set of user attributes A_U such that A_U satisfies the policy P. An example is a policy that asks for a user's year of birth but the user is over 100 years old. The resulting anonymity set can be very small.

7.3 Specific definitions of unlinkability

7.3.1 General

While the terms “unlinkable” or “anonymous” are often used, the meaning actually varies depending on the context of the use of the terms. To speak about them precisely, it is necessary to speak of from which adversary role, what target entity role is unlinkable.

This document adopts a naming convention for notions of unlikability where the adversarial role constitutes the first part of the name and the second part describes the target entity role. The two parts are separated by a “-”, as shown in ~~Figure 2~~.



Key

1 adversary

2 target entity

[a] Unlinkability.

Figure 2 — Naming convention for notions of unlinkability

This document provides the following notions of unlinkability for attribute-based entity authentication protocols, indicating this aspect together with “unlinkability”.

1) Passive outsider unlinkability (PO-U). The adversary plays none of the U, RP or AP roles but can passively observe the content of all exchanged messages. The target entity role is U.

2) Active outsider unlinkability (AO-U). The adversary plays none of the U, RP or AP roles but can observe, actively intercept and modify exchanged messages in a man-in-the-middle fashion. The target entity role is U.

3) RP-U unlinkability. The adversary can observe, actively intercept and modify exchanged messages and additionally plays the role of RP. The target entity role is U.

4) AP-U unlinkability. The adversary can observe, actively intercept and modify exchanged messages and additionally plays the role of AP. The target entity role is U.

5) RP+AP-U unlinkability. The adversary can observe, actively intercept and modify exchanged messages and additionally plays the role of both RP and AP. The target entity role is U.

6) AP-RP unlinkability. The adversary can observe, actively intercept and modify exchanged messages and additionally plays the role of AP. The target entity role is RP.

7) AP-RP+U unlinkability. The adversary can observe, actively intercept and modify exchanged messages and additionally plays the role of AP. The target entity roles are U and RP. This means that neither U nor RP can be linked.

8) RP+RP'-U unlinkability. The adversary can observe, actively intercept, and modify exchanged messages and additionally plays the roles of RP and RP'. The target entity role is U. In this case, unlinkability of U by RP is not in question. Here, a different type of linking is considered. That is, linking as the ability for RP and RP’ to distinguish:

   — two protocol executions between U and RP and U and RP’ from;

   — two protocol executions between U and RP and U’ and RP’.

~~Table 1~~ summarizes, for each notion of unlinkability, the settings for the adversary and target entity roles.

Table 1 — Relationship between adversarial and target roles and notions of unlinkability

Notion of unlinkability	Adversarial role(s)	Target role(s)	Explanations
Passive outsider (PO-U)	PO	U	Attempt to track U across authentications while these are being monitored (read-only).
Active outsider (AO-U)	AO	U	Attempt to track U across authentications while these are being controlled (read-write).
RP-U	RP	U	The RP attempts to track the U across authentications

Table 1 *(continued)*

Notion of unlinkability	Adversarial role(s)	Target role(s)	Explanations
AP-U	AP	U	The AP attempts to track the U across authentications.
RP+AP-U	RP and AP	U	The colluding RP and AP attempt to track U across authentications.
AP-RP	AP	RP	The AP attempts to track the RP across authentications.
AP-RP+U	AP	RP and U	The AP attempts to track U, RP, and the pair (U, RP) across authentications.
RP+RP'-U	RP and RP'	U	Colluding RPs attempt to track the U across authentications. RP may be able to track U in transaction with RP, but cannot track the same U communicating with RP'

For each of the roles controlled by the adversary, the adversary may arbitrarily deviate from the protocol specification in attempts to defeat the unlinkability.

It is common that after a successful authentication, a session is held between U and RP. The session management ensures that during the session, it is same U that is talking to RP, thus it is linkable within the session. In this document, the notion of unlinkability discussed is between the sessions and not within.

~~Annex A~~ describes each of the unlinkability notions in more detail.

7.3.2 Passive outsider unlinkability (anti-tracking from passive outsiders)

An attribute-based authentication protocol is said to achieve passive outsider unlinkability if an adversary that is only allowed to observe the exchanges among U, AP and RP cannot link these observations to U. For example, when Alice, assuming the role of U, authenticates herself to RP via the help of the AP, if the adversary cannot link the run of the protocol to Alice, passive outsider unlinkability is satisfied.

Achieving this notion of unlinkability is not considered technically difficult. Usually, passive observations can be neutralised by careful use of encryption throughout the protocol.

7.3.3 Active outsider unlinkability (anti-tracking from active outsiders)

Unlike in the passive outsider unlinkability case, the adversary now can intercept the message exchanges among U, AP and RP and can potentially modify them. In other words, the adversary is an entity that remains external to all parties but has read-write access to the contents of the messages exchanged at each stage of the protocol. In particular, the adversary may carry out man-in-the-middle attacks while parties are interacting as per the protocol. The adversary attempts to link a protocol run to U and active outsider unlinkability is satisfied when it is shown that this is not possible.

Achieving this notion of active outsider unlinkability is not considered technically difficult. Active outsiders can be neutralized by carefully using the correct combination of existing encryption and authentication throughout the protocol.

7.3.4 RP-U unlinkability ("anonymous visits" to an RP)

This is a common notion of user anonymity towards RP. The RP cannot tell whether the user-agent U that arrived at the RP has visited it before. If RP-U unlinkability is not satisfied and RP can link visits made by the same user-agent, then U is not anonymous from the point of view of the RP but is only pseudonymous. RP-U unlinkability is satisfied when an adversary that assumes the role of RP and has also read-write access to all transmissions between U, RP and AP, cannot link U across authentications.

Satisfying RP-U unlinkability guaranties that U is protected by a strong form of anonymity towards RP when performing successive authentications.

7.3.5 AP-U unlinkability

In this notion of unlinkability, the AP attempts to tell whether the user-agent U that arrived at the RP has visited it before. If AP-U unlinkability is not satisfied and AP can link visits made by the same user-agent, then U is not anonymous from the point of view of the AP but is only pseudonymous. AP-U unlinkability is satisfied when an adversary that assumes the role of AP and has also read-write access to all transmissions between U, RP and AP, cannot link U across authentications.

Satisfying AP-U unlinkability guaranties that U is protected by a strong form of anonymity towards AP when performing successive authentications.

7.3.6 RP+AP-U unlinkability (anti-RP-AP-collusion)

In this notion of unlinkability, RP and AP collude, i.e. share all their resources, in an attempt to link authentications performed by the same user-agent. The RP-AP collusion has also read-write access to all transmissions between U, RP and AP. If the authentication protocol protects the user from this kind of attack, RP+AP-U unlinkability is said to be achieved.

An attribute-based protocol that satisfies RP+AP-U unlinkability provides the strongest possible form of anonymity for the user, as the user remains anonymous even if the rest of the universe is adversarial.

7.3.7 AP-RP unlinkability (anti-tracking of RP from AP)

In this notion of unlinkability, the AP attempts to tell whether the same RP is involved in multiple authentications. If AP-RP unlinkability is not satisfied and AP can link user authentications made to the same RP, then RP is not anonymous from the point of view of the AP but is only pseudonymous. AP-RP unlinkability is satisfied when an adversary that assumes the role of AP and has also read-write access to all transmissions between U, RP and AP, cannot link RP across authentications.

Satisfying AP-RP unlinkability guaranties that RP remains anonymous towards AP across successive authentications.

7.3.8 AP-RP+U unlinkability

Here, the AP attempts to tell whether the same pair of entities (U, RP) is involved in multiple authentications. Beyond owning AP, the adversary has also read-write access to all exchanges between U, RP and AP. This notion is relevant because it is not necessarily implied by the conjunction of AP-U unlinkability and AP-RP unlinkability. Indeed, the protocol can involve variables that are invariant across authentications for a given pair (U, RP).

AP-RP+U unlinkability guaranties not only anonymity of RP and U respectively, but also that the pair of entities (U, RP) is jointly protected by some form of anonymity towards AP when U and RP engage in successive authentications.

7.3.9 RP+RP'-U unlinkability (anti-tracking of U from a set of colluding RPs)

In this notion of unlinkability, a set of colluding RPs attempt to link two authentication events performed by the same U, one with RP and the other one with RP'. For example, if the same identifier is used for the user across the RPs or some correlation functions (e.g. "cookie-sync") exist, this attack becomes trivial.

When U is in a large anonymity set, the attack can be mitigated by using different identifiers to different RPs and protecting the runs of protocol from the correlation functions. Such identifiers are often called pairwise-pseudonymous identifiers (PPID) and can be achieved even if RP-U unlinkability is not achieved. However, with only PPID, it is difficult to mitigate the correlation functions such as "cookie-sync" implemented by the set of colluding RPs. By achieving RP-U unlinkability where RP here is taken as a set of RPs achieves RP+RP'-U unlinkability.

7.4 Relationships between notions of unlinkability

Some of the notions of unlinkability defined in ~~7.3~~ are implied by others in a generic sense, i.e. regardless of the specificities of the attribute-based entity authentication protocol at hand. It can be shown that:

— Active Outsider unlinkability implies Passive Outsider unlinkability;

— RP-U unlinkability implies Active Outsider unlinkability;

— AP-U unlinkability implies Active Outsider unlinkability;

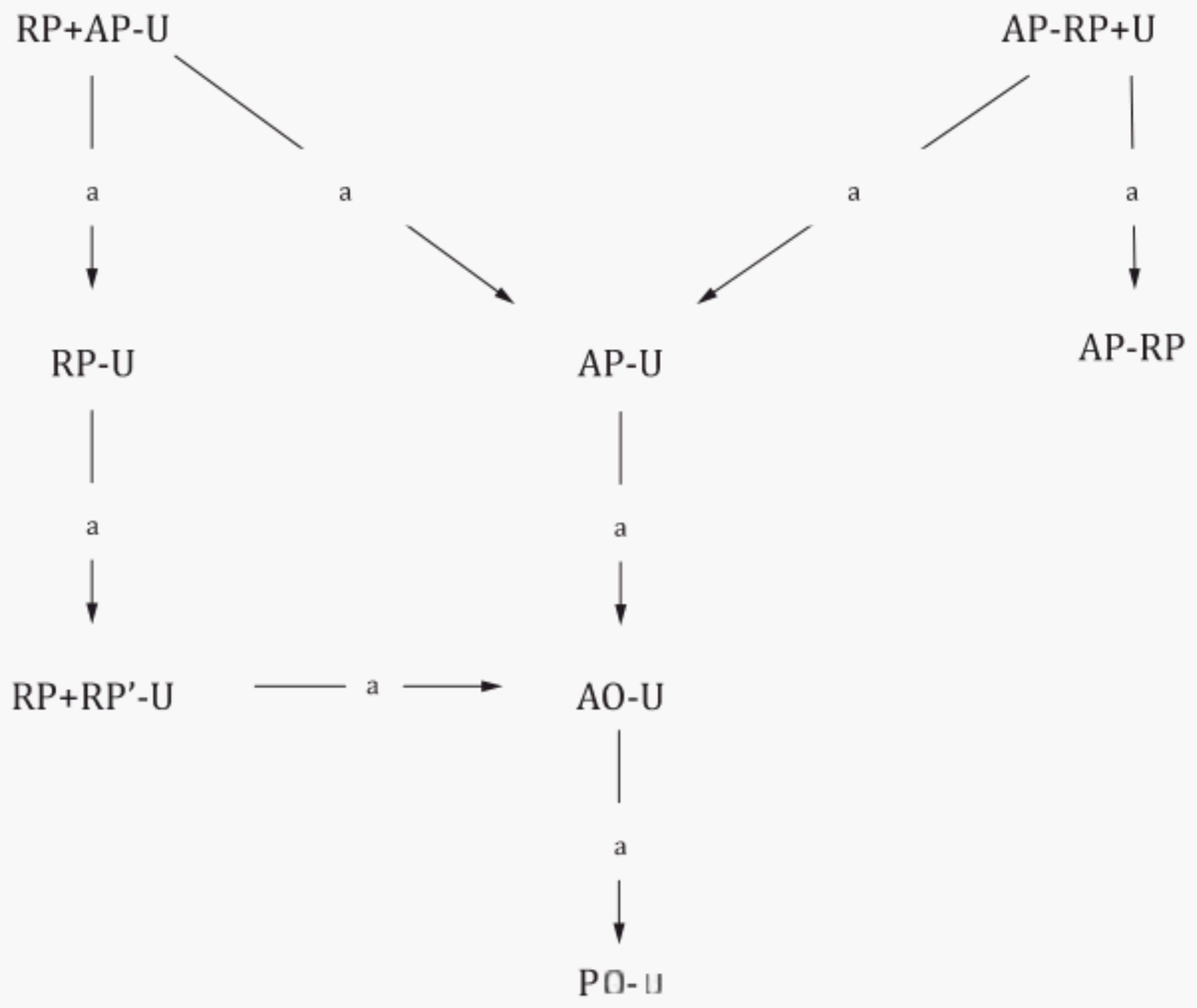
— RP+AP-U unlinkability implies RP-U unlinkability;

— RP+AP-U unlinkability implies AP-U unlinkability;

— AP-RP+U unlinkability implies AP-RP unlinkability;

— AP-RP+U unlinkability implies AP-U unlinkability.

~~Figure 3~~ depicts these relationships.



[a] Implies.

Figure 3 — Implications between unlinkability properties for attribute-based entity authentication protocols

7.5 Unlinkability levels for attribute-based entity authentication

~~Subclause 7.5~~ defines a simplified metric to measure the unlinkability properties of an ABUEA protocol. This metric makes it possible to express how ABUEA protocols protect the anonymity of involved parties and to establish classes of reference for ABUEA protocols.

The metric relies on a scale of increasingly strong unlinkability levels (UL). A UL is made of an integer ranging from 1 to 5 indicating how well the attribute-based entity authentication protocol protects the anonymity of U. 1 represents the basic protection against passive observation whereas 5 indicates the maximal protection provided by RP+AP-U unlinkability.

Since RP-U unlinkability and AP-U unlinkability are independent and incomparable properties, the unlinkability level 3 does not differentiate between them. The protocol shall indicate which unlinkability is supported. Unlinkability level 4 requires attaining RP-U and AP-U unlinkability.

~~Table 2~~ lists the unlinkability properties required from an attribute-based entity authentication protocol when claiming a particular UL.

Table 2 — Unlinkability levels and the properties required for each level

Unlinkability level (UL)	Required unlinkability properties
1	PO-U unlinkability
2A	AO-U unlinkability
2B	RP+RP'-U unlinkability
3A	RP-U unlinkability
3B	AP-U unlinkability
4	RP-U unlinkability and AP-U unlinkability
5	RP+AP-U unlinkability

Also, a UL is marked with "+" or "++" when the protocol offers some protection against the tracking of RP, which is viewed as a secondary objective of the protocol. The notation "+" indicates that the protocol achieves AP-RP unlinkability while "++" indicates that it achieves AP-RP+U unlinkability. As an example, UL 3++ is a stronger UL than UL 3+, which is itself stronger than UL 3.

The unlinkability levels and their required properties are instrumental to privacy impact assessment privacy controls rationale and to related risk assessment scale, i.e. for the purposes of risk evaluation, likelihood and mitigation approach. Such levels can be adapted to other privacy principles.

7.6 Models

ABUEA models that extend beyond the minimal U-RP-AP model exist. These models may define additional entity roles or make use of a specific terminology to refer to them. ~~Annex B~~ describes examples and analyses of protocols and their unlinkability levels and properties. ~~Annex C~~ provides examples and analyses of specific implementations of ABUEA.

8 Attributes

8.1 Categories of attributes

8.1.1 Personal attributes

Personal attributes are the attributes of the person that describes certain aspect of the person.

EXAMPLE Name, age, location, blood pressure, domicile address, employer, phone number, e-mail address, etc.

8.1.2 Self-claimed attributes

Self-claimed attributes are the attributes provided by the person without any proof.

8.1.3 Verified attributes

Verified attributes are the personal attributes that are verified by the registration authority when the person registered at the AP.

8.1.4 Static attributes

Static attributes are the attributes that do not change.

EXAMPLE The name at birth or date of birth does not change.

8.1.5 Semi-static attributes

Semi-static attributes are the attributes that are stable for a long period of time but can potentially change.

EXAMPLE The family name changes at marriage in certain cultures. In this case, the family name is a semi-static attribute.

8.1.6 Dynamic attributes

Dynamic attributes change often.

EXAMPLE The GPS coordinate of the person changes quite frequently.

8.1.7 Computed attributes

Computed attributes are attributes that are computed from one or more of the attributes.

EXAMPLE An attribute "is_over_13" whose value is "true" or "false" is computed from the date in question and the date of birth of the person.

8.1.8 Identifying attributes

Identifying attributes are attributes that contributes to uniquely identifying a user within a context.

8.1.9 Supporting attributes

Supporting attributes are attributes that is used in identity proofing but not as an identifying attribute.

8.2 Verified attribute expiry and revocation

For dynamic and semi-static attributes and the computed attributes that rely on them, the validity of the attribute values is time-bound. Thus, the attribute may need to be associated with the metadata that indicates the validity period such as expiry date or provided with another form of service that assists with the verification of the validity of the data, unless such verification is unnecessary for the receiving service.

In the absence of such associated metadata, there shall be some form of revocation service supplied in the system. However, some implementation of such services can cause the RP to leak the information to AP that the RP indeed has received information about the U thus breaking AP-RP unlinkability.

8.3 Attribute assurance

For attribute authentication, level of assurance of the relevant attribute directly affects the level of authentication. ISO/IEC TS 29003 provides more information on the concept.

9 Requirements for level N attribute-based unlinkable entity authentication

To attain unlinkability level N attribute-based entity authentication, the protocol shall:

a) be correct;

b) be unforgeable;

c) satisfy the assurance level on attributes that is required by the RP; and

d) satisfy the unlinkability properties at level N.

Annex A
(informative)

Formal definitions for security and unlinkability notions

A.1 General

This annex provides formal definitions for the notions of unforgeability and the different notions of unlinkability considered in this document.

Each notion is defined based on a probabilistic experiment called a game wherein the adversary A and the target entity or entities play their roles as per the specification of the considered protocol. The game defines a goal and resources for the adversary as well as a winning event for the adversary. Each notion defines its own game, which is meant to capture the real-life attack scenario considered by the notion.

A.2 Unforgeability

Unforgeability captures the fact that U should be unable to successfully authenticate towards RP when the set of attributes attached to U does not satisfy the policy P imposed by RP.

The game is defined as follow. Arbitrary entities play the role of AP and RP while the adversary A plays the role of U at each stage of the protocol. A is subjected to the game where, sequentially:

1) AP and RP execute the setup phase (if any);
2) AP and A playing the role of U execute the user registration phase (if any). A set of attributes A_U is agreed upon and attached to U during that phase;
3) RP, AP and A playing the role of U execute the authentication phase for a policy P that is not satisfied by A_U;
4) The winning event for the adversary is defined as a successful authentication towards RP.

The protocol achieves unforgeability if it is unfeasible for the adversary A to win, that is, to successfully authenticate in these conditions, with a probability significantly better than negligible.

Unforgeability ensures that U performs user registration and be attached a set of attributes that matches the policy in order to successfully perform authentications towards RP.

A.3 Passive Outsider (PO-U) unlinkability

This clause formally defines PO-U unlinkability by describing the concrete adversarial scenario that the protocol resists to achieve this level of unlinkability.

Simply put, the authentication protocol is said to achieve PO-U unlinkability if the adversary that can observe the contents of all protocol exchanges among U, AP and RP, cannot link its observations to U.

The adversary A is an entity that remains external to all parties but has read-only access to the contents of the messages exchanged at each stage of the protocol. This includes, in sequence:

1) the adversary A chooses the set of attributes for U_0 and U_1 as well as policy P;

2) the messages exchanged between AP and RP during the setup phase (if any);

3) the messages exchanged between AP and a first entity U_0 playing the role of U in the user registration phase (if any);

4) the messages exchanged between AP and a second entity U_1 playing the role of U in the user registration phase (if any). U_1 and U_0 are arbitrary distinct entities but the sets of attributes attached to them both fulfil the policy P;

5) the messages exchanged between RP, AP and U_0 in the authentication phase;

6) the messages exchanged between RP, AP and an unknown entity U_b playing the role of U in the authentication phase, where the bit b can either be 0 or 1 with equal probabilities;

7) the adversary A returns a guess $b' \in \{0,1\}$ on the value of b. The winning event for the adversary is defined as $b' = b$.

The protocol achieves PO-U unlinkability if it is unfeasible for the adversary to win, that is, to make a correct guess, with a probability significantly better than one half.

NOTE Unfeasible in the context of ~~A.3~~ to ~~A.10~~, refers to computationally bounded adversaries (at least for most constructions).

Achieving this level of unlinkability is not considered technically difficult. Usually, passive observations can be neutralized by careful use of encryption throughout the protocol.

A.4 Active Outsider (AO-U) unlinkability

Unlike in the case of PO-U unlinkability, the adversary is now allowed to modify the contents of the messages exchanged between U, AP and RP in addition to observing them. The rest of the game remains unchanged.

The adversary A is an entity that remains external to all parties but has read-write access to the contents of the messages exchanged at each stage of the protocol. This includes, in sequence:

1) the adversary A chooses the set of attributes for U_0 and U_1 as well as policy P. The messages exchanged between AP and RP during the setup phase (if any);

2) the messages exchanged between AP and a first entity U_0 playing the role of U in the user registration phase (if any);

3) the messages exchanged between AP and a second entity U_1 playing the role of U in the user registration phase (if any). U_1 and U_0 are arbitrary distinct entities but the sets of attributes attached to them both fulfil the policy P;

4) the messages exchanged between RP, AP and U_0 in the authentication phase;

5) the messages exchanged between RP, AP and an unknown entity U_b playing the role of U in the authentication phase, where the bit b can either be 0 or 1 with equal probabilities;

6) the adversary A returns a guess $b' \in \{0,1\}$ on the value of b. The winning event for the adversary is defined as $b' = b$.

The protocol achieves AO-U unlinkability if it is unfeasible for the adversary to win, that is, to make a correct guess, with a probability significantly better than one half.

Achieving this level of unlinkability is not considered technically difficult. Active outsiders can be neutralized by carefully using the correct combination of existing encryption and authentication throughout the protocol.

A.5 RP-U unlinkability

This is a common notion of "anonymity" at the RP, i.e. the RP cannot tell if the user that arrived at the RP has visited it before. If RP-U unlinkability is not satisfied and RP can link two visits made by the same user, then the protocol is not anonymous from the point of view of the RP but is only pseudonymous.

The game is defined as follow. The adversary A plays the role of RP at each stage of the protocol and has read-write access to the contents of all exchanged messages between the parties. A is subjected to the game where, sequentially:

1) the adversary A chooses the set of attributes for U_0 and U_1 as well as policy P;
2) AP and RP execute the setup phase (if any);
3) AP and a first entity U_0 playing the role of U execute the user registration phase (if any);
4) AP and a second entity U_1 playing the role of U execute the user registration phase (if any). U_1 and U_0 are arbitrary distinct entities but the sets of attributes attached to them both fulfil the policy P;
5) RP, AP and U_0 execute the authentication phase;
6) RP, AP and an unknown entity U_b execute the authentication phase, where b can either be 0 or 1 with equal probabilities;
7) the adversary A returns a guess $b' \in \{0,1\}$ on the value of b. The winning event for the adversary is defined as $b' = b$.

The protocol achieves RP-U unlinkability if it is unfeasible for the adversary A to win, that is, to make a correct guess, with a probability significantly better than one half.

RP-U unlinkability ensures that U is protected by some form of anonymity towards RP when performing successive authentications.

A.6 AP-U unlinkability

The game is defined as follow. The adversary A plays the role of AP at each stage of the protocol and has read-write access to the contents of all the messages exchanged between the parties. A is subjected to the game where, sequentially:

1) the adversary A chooses the set of attributes for U_0 and U_1 as well as policy P;
2) AP and RP execute the setup phase (if any);
3) AP and a first entity U_0 playing the role of U execute the user registration phase (if any);
4) AP and a second entity U_1 playing the role of U execute the user registration phase (if any). U_1 and U_0 are arbitrary distinct entities but the sets of attributes attached to them both fulfil the policy P;
5) RP, AP and U_0 execute the authentication phase;
6) RP, AP and an unknown entity U_b execute the authentication phase, where b can either be 0 or 1 with equal probabilities;
7) the adversary A returns a guess $b' \in \{0,1\}$ on the value of b. The winning event for the adversary is defined as $b' = b$.

The protocol achieves AP-U unlinkability if it is unfeasible for the adversary A to win, that is, to make a correct guess, with a probability significantly better than one half.

AP-U unlinkability ensures that U is protected by some form of anonymity towards AP when performing successive authentications.

A.7 RP+AP-U unlinkability

This notion captures the case when RP and AP collude against U.

The game is defined as follow. The adversary A plays the role of both RP and AP at each stage of the protocol. This implies that A has read-write access to the contents of all the messages exchanged between the parties.

A is subjected to the game where, sequentially:

1) the adversary A chooses the set of attributes for U_0 and U_1 as well as policy P;
2) AP and RP execute the setup phase (if any);
3) AP and a first entity U_0 playing the role of U execute the user registration phase (if any);
4) AP and a second entity U_1 playing the role of U execute the user registration phase (if any). U_1 and U_0 are arbitrary distinct entities but the sets of attributes attached to them both fulfil the policy P;
5) RP, AP and U_0 execute the authentication phase;
6) RP, AP and an unknown entity U_b execute the authentication phase, where b can either be 0 or 1 with equal probabilities;
7) the adversary A returns a guess $b' \in \{0,1\}$ on the value of b. The winning event for the adversary is defined as $b' = b$.

The protocol achieves RP+AP-U unlinkability if it is unfeasible for the adversary A to win, that is, to make a correct guess, with a probability significantly better than one half.

RP+AP-U unlinkability ensures that U is protected by the strongest possible form of anonymity when performing successive authentications. The actions of U remain unlinkable even when RP and AP collude.

A.8 AP-RP unlinkability

This notion captures the case when AP is attempting to link authentications to the same RP.

The game is defined as follow. The adversary A plays the role of AP at each stage of the protocol and has read-write access to the contents of all messages exchanged between the parties. A is subjected to the game where, sequentially:

1) AP and a first entity RP_0 playing the role of RP execute the setup phase (if any);
2) AP and a second entity RP_1 playing the role of RP execute the user registration phase (if any). RP_1 and RP_0 are arbitrary distinct entities with the same policy P;
3) AP and U execute the user registration phase (if any);
4) AP, U and RP_0 execute the authentication phase;
5) AP, U and RP_b execute the authentication phase, where b can either be 0 or 1 with equal probabilities;
6) the adversary A returns a guess $b' \in \{0,1\}$ on the value of b. The winning event for the adversary is defined as $b' = b$.

The protocol achieves AP-RP unlinkability if it is unfeasible for the adversary A to win, that is, to make a correct guess, with a probability significantly better than one half.

AP-RP unlinkability ensures that RP is protected by some form of anonymity towards AP when performing successive authentications.

A.9 AP-RP+U unlinkability

This notion captures the case when AP is attempting to link authentications to the same pair of entities (RP, U).

The game is defined as follow. The adversary A plays the role of AP at each stage of the protocol and has read-write access to the contents of all messages exchanged between the parties. A is subjected to the game where, sequentially:

1) the adversary A chooses the set of attributes for U_0 and U_1 as well as policy P.
2) AP and a first entity RP_0 playing the role of RP execute the setup phase (if any);
3) AP and a second entity RP_1 playing the role of RP execute the user registration phase (if any). RP_1 and RP_0 are arbitrary distinct entities with the same policy P;
4) AP and a first entity U_0 playing the role of U execute the user registration phase (if any);
5) AP and a second entity U_1 playing the role of U execute the user registration phase (if any). U_1 and U_0 are arbitrary distinct entities but the sets of attributes attached to them both fulfil the policy P;
6) AP, U_0 and RP_0 execute the authentication phase;
7) AP, U_b and RP_b execute the authentication phase, where b can either be 0 or 1 with equal probabilities;
8) the adversary A returns a guess $b' \in \{0,1\}$ on the value of b. The winning event for the adversary is defined as $b' = b$.

The protocol achieves AP-RP+U unlinkability if it is unfeasible for the adversary A to win, that is, to make a correct guess, with a probability significantly better than one half.

AP-RP+U unlinkability ensures that the pair of entities (RP, U) is protected by some form of anonymity towards AP when engaging into successive authentications.

A.10 RP+RP'-U unlinkability (anti-tracking of U from a set of colluding RPs)

This is a common notion of "anti-tracking" across the colluding RPs, i.e. the RPs cannot tell if the user that arrived at the RP' is the same user that has visited RP before. If RP+RP'-U unlinkability is not satisfied and RPs can link two visits made by the same user, then the protocol is not unlinkable from the point of view of the RPs.

The game is defined as follow. The adversary A plays the role of colluding RPs (RP and RP') at each stage of the protocol and has read-write access to the contents of all exchanged messages between the parties. A is subjected to the game where, sequentially:

1) the adversary A chooses the set of attributes for U_0 and U_1 as well as policy P;
2) AP and RP, RP' execute the setup phase (if any);
3) AP and a first entity U_0 playing the role of U execute the user registration phase (if any);
4) AP and a second entity U_1 playing the role of U execute the user registration phase (if any). U_1 and U_0 are arbitrary distinct entities but the sets of attributes attached to them both fulfil the policy P;
5) RP, AP and U_0 execute the authentication phase;
6) RP', AP and an unknown entity U_b execute the authentication phase, where b can either be 0 or 1 with equal probabilities;
7) the adversary A returns a guess $b' \in \{0,1\}$ on the value of b. The winning event for the adversary is defined as $b' = b$.

The protocol achieves RP+RP'-U unlinkability if it is unfeasible for the adversary A to win, that is, to make a correct guess, with a probability significantly better than one half.

RP+RP'-U unlinkability ensures that U is protected by some form of anonymity towards colluding RPs when performing successive authentications.

Annex B
(informative)

Examples of attribute-based entity authentication protocols

B.1 General

This annex provides examples of attribute-based entity authentication protocols. These protocols are not meant to be implemented in real-life applications but to be shown as examples of typical constructions for attribute-based authentication. The examples show how to determine what security and unlinkability properties they achieve and to which UL they correspond.

B.2 Protocol 1

B.2.1 General

This clause describes a simple example of an attribute-based entity authentication protocol that provably fulfils the properties of correctness and unforgeability. It belongs to the class UL 0+.

The protocol features preliminary phases (a setup phase and a user registration phase) that are carried out before the authentication phase itself.

B.2.2 Setup phase

1) AP generates an asymmetric key pair (sk_{AP}, vk_{AP}) for a digital signature mechanism.

2) RP obtains the public verification key vk_{AP} of AP.

B.2.3 User registration phase

1) U generates an asymmetric key pair (sk_U, vk_U) for a digital signature mechanism.

2) U sends its public verification key vk_U to AP.

3) U and AP jointly agree on a set of attributes $A_U = \{a_1, \ldots, a_n\}$ that is attached to U.

4) AP provides U with a signature s_i on each pair (vk_U, a_i) for $i = 1, \ldots, n$.

B.2.4 Authentication phase

1) U sends an authentication request to RP.

2) RP sends an attribute policy P to U as well as a random number r.

3) U:

   a) Parses P to identify the subset $I \subseteq \{1, \ldots, n\}$ of attributes involved in the policy.

   b) Collects the involved attributes and their signature as $L = \{(a_i, s_i) \mid i \in I\}$.

   c) Uses its signing key sk_U to generate a signature s on (L, r).

d) Sends *proof* = (vk_U, L, s) to RP.

4) RP:

a) Parses *proof* to recover vk_U, L and s.

b) Uses the verification key vk_U to verify that s is a valid signature of U on (L, r).

c) Uses the verification key vk_{AP} to verify that s_i is a valid signature of AP on (vk_U , a_i) for $i \in I$.

d) Verifies that $P(\{a_i \mid i \in I\}) = true$. If so then U is successfully authenticated towards RP. Otherwise, or if any of the above steps fails for any reason, then authentication fails.

B.2.5 Analysis and UL classification

This example of attribute-based entity authentication protocol enjoys:

a) correctness: this is easily checked from the description of the protocol;

b) unforgeability: observing the unforgeability game, it is easily seen that in order to pass the attribute verification step 4) d), U needs to change the value of at least one attribute a_i. This implies that U needs to replace s_i by a valid signature on the changed value of a_i. This cannot be done since only AP possesses the signing key sk_{AP};

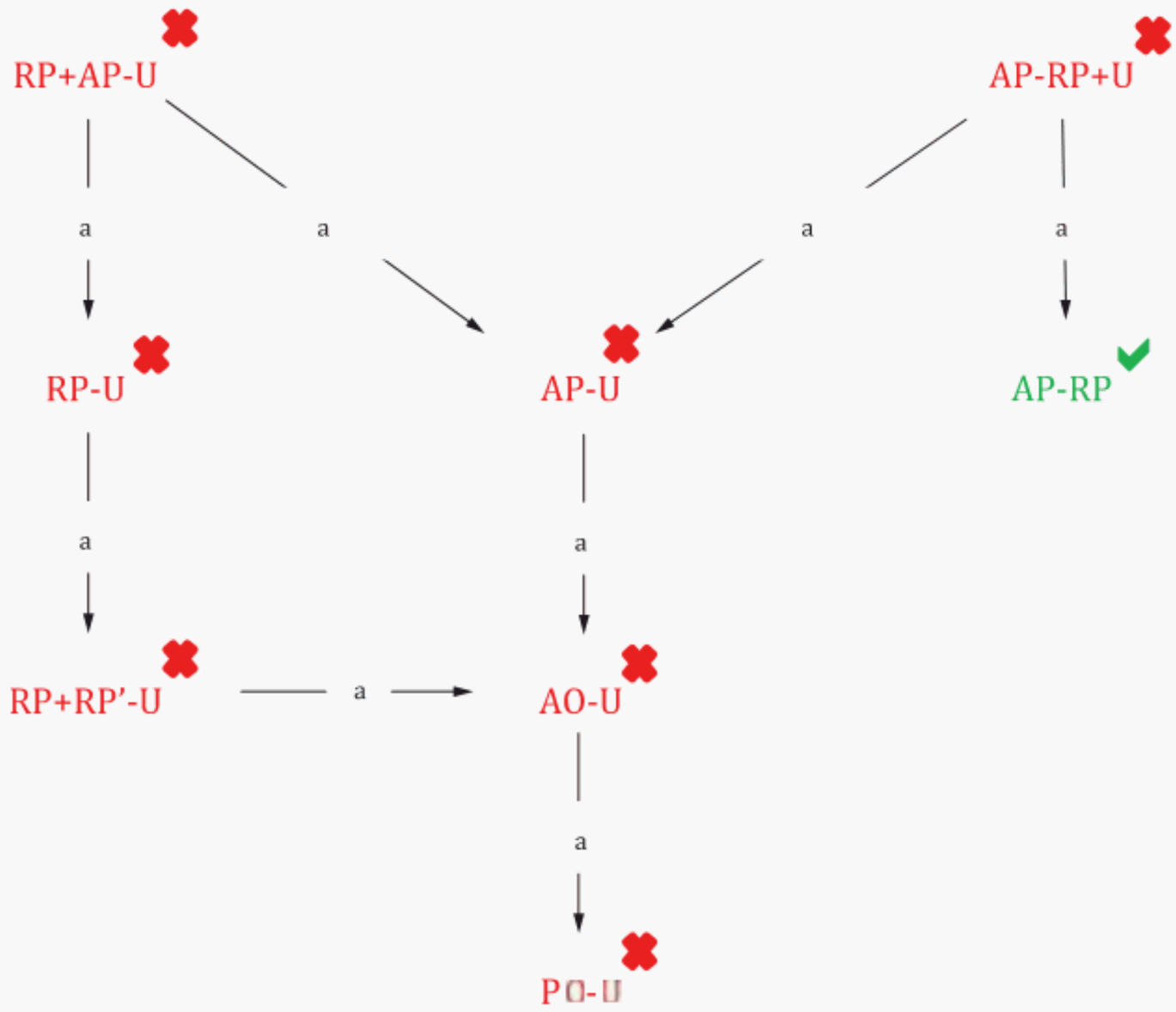
NOTE It can be proved formally that a successful adversary in the unforgeability game can be turned into an efficient attack against the signature scheme used by AP. Assuming that this signature scheme is existentially unforgeable, the adversary fails with overwhelming probability.

However, carrying out an authentication fully reveals to RP all the attributes attached to U that are involved in the policy. Also, Protocol 1 leaves all communications unprotected against eavesdropping. A passive observation of the contents exchanged between U and RP fully reveals the public key vk_U. Consequently, authentications involving U can be linked and Protocol 1 is not PO-U unlinkable.

Since RP-U, AP-U, RP+AP-U and AP-RP+U unlinkability would all imply PO-U unlinkability, Protocol 1 does not enjoy any of these properties either.

Finally, one sees that RP never sends identifiable information in the setup or authentication phases. Protocol 1 is therefore AP-RP unlinkable.

Our findings are summarized on ~~Figure B.1~~.



[a] Implies.

Figure B.1 — Unlinkability properties achieved by Protocol 1.

Applying ~~Table 2~~, Protocol 1 belongs to the class UL 0+.

B.3 Protocol 2

B.3.1 General

Protocol 2 is the same as Protocol 1 but adds some resistance against active outsiders. It belongs to the class UL 2.

B.3.2 Description

Protocol 2 is identical to Protocol 1 with the following additional operations.

In the setup stage, RP generates an additional key pair (ek_{RP}, dk_{RP}) for an asymmetric encryption scheme and receives from AP a signature *sig* on its public encryption key ek_{RP} under the signing key sk_{AP}.

At Step 2 of the authentication phase, RP additionally sends (ek_{RP}, *sig*) to U.

At Step 3) d), U verifies that *sig* is a valid signature on ek_{RP} using the verification key vk_{AP} of AP and sends *proof* encrypted under ek_{RP} to RP instead of just its cleartext value. U aborts if *sig* is invalid.

At Step 4) a), RP starts by decrypting *proof* using the decryption key dk_{RP} and then proceeds as in Protocol 1.

B.3.3 Analysis and UL classification

Protocol 2 enjoys:

a) correctness: identical to Protocol 1;

b) unforgeability: identical to Protocol 1;

c) PO-U unlinkability: since the contents of *proof* are now encrypted and only RP possesses the decryption key dk_{RP}, no information can be inferred on U through passive observation.

NOTE 1 It can be formally established that if the encryption scheme is semantically secure, then the PO-U adversary has a winning probability negligibly close to one half.

d) AO-U unlinkability: the adversary may attempt a man-in-the-middle attack by replacing the encryption key of RP with its own to be able to decrypt *proof* and determine which of vk_{U0} or vk_{U1} appears in *proof*. However, since only AP knows sk_{AP}, this would require the forgery of a valid signature on the adversary's public key. This is infeasible assuming that the signature scheme is secure.

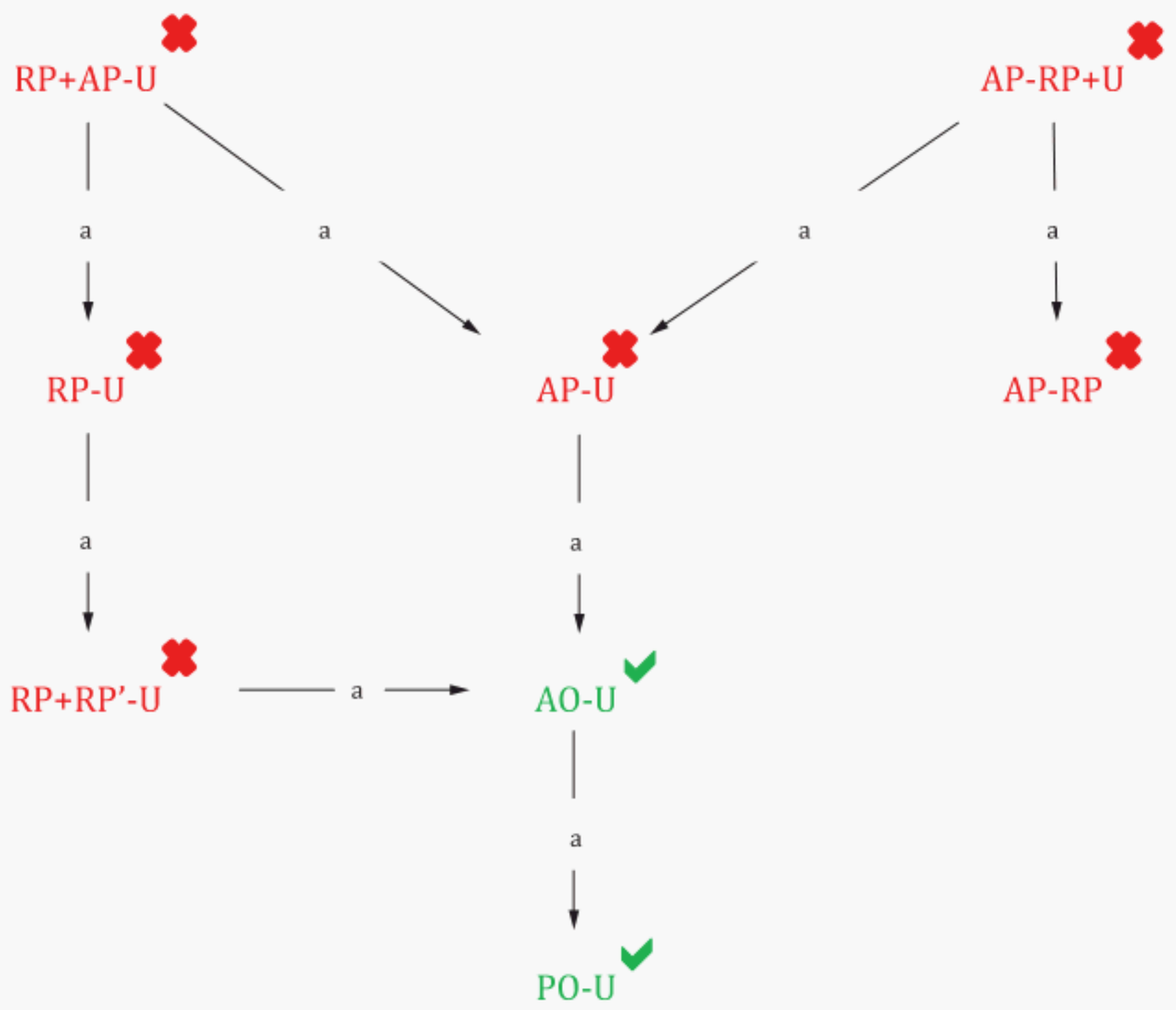
NOTE 2 It can be formally established that if the signature scheme is existentially unforgeable, then the AO-U adversary has a winning probability negligibly close to one half.

However, Protocol 2 is not RP-U unlinkable since U reveals its public verification key vk_U to RP during the authentication phase.

Protocol 2 is not AP-U unlinkable either; looking at the AP-U game, one sees that it is easy for the AP-U adversary (which controls AP) to actively replace the pair (ek_{RP}, *sig*) sent by RP to U_b at Step 2 of authentication with another pair (*ek*, *sig'*) where *ek'* is a random public encryption key and *sig'* a signature on *ek'* under sk_{AP}. The AP-U adversary then reads the encryption of proof returned by U_b and decrypts it to recover *proof*. Given *proof*, the adversary easily determines which of vk_{U0} or vk_{U1} is involved and guesses *b* with probability one.

By looking at the AP-RP game, one also sees that the adversary easily deduces from what RP sends to U at Step 2 which of ek_{RP0} or ek_{RP1} is being used and correctly guesses *b* with probability one. Hence, Protocol 2 is not AP-RP unlinkable.

Our findings are summarized on ~~Figure B.2~~.



[a] Implies.

Figure B.2 — Unlinkability properties achieved by Protocol 2.

Applying ~~Table 2~~, Protocol 2 belongs to the class UL 2.

B.4 Protocol 3

B.4.1 General

Protocol 3 provides an example of an attribute-based authentication protocol that achieves RP-U unlinkability. The protocol belongs to class UL 3+.

The protocol features preliminary phases (a setup phase and a user registration phase) that are carried out before the authentication phase itself.

B.4.2 Setup phase

1) AP generates an asymmetric key pair (sk_{AP}, vk_{AP}) for a digital signature mechanism.
2) AP generates an asymmetric key pair (ek_{AP}, dk_{AP}) for an encryption mechanism.
3) RP obtains the public verification key vk_{AP} of AP.

B.4.3 User registration phase

1) AP sends its public keys (vk_{AP}, ek_{AP}) to U as well as a signature s_{AP} on ek_{AP} under sk_{AP}.
2) U uses the verification key vk_{AP} of AP to verify that s_{AP} is a valid signature on ek_{AP}. 3) U generates an asymmetric key pair (sk_U, vk_U) for a digital signature mechanism.
4) U sends its public verification key vk_U to AP.
5) U and AP jointly agree on a set of attributes $A_U = \{a_1, \ldots, a_n\}$ that will be attached to U.
6) AP records (vk_U, A_U) in a database.

B.4.4 Authentication phase

1) U sends an authentication request to RP.
2) RP sends an attribute policy P to U as well as a random number *r*.
3) U:
   a) Uses its signing key sk_U to generate a signature *s* on (P, *r*).
   b) Encrypts (P, *r*, *s*, vk_U) under the encryption key ek_{AP} of AP to obtain a ciphertext *c*.
   c) Sends *c* to AP.
4) AP:
   a) Uses its decryption key dk_{AP} to decrypt *c* and parses the resulting plaintext as (P, *r*, *s*, vk_U).
   b) Searches for a record (vk_U, A_U) in its database.
   c) Uses vk_U to verify that *s* is a valid signature of U on (P, *r*).
   d) Asserts that $P(A_U) = true$.
   e) Uses its signing key sk_{AP} to generate a signature *sig* on (P, *r*).
   f) Sends *sig* to U. If any of the above steps fails for any reason, then AP aborts.
5) U forwards *sig* to RP.

6) RP:

   a) Uses the verification key vk_{AP} of AP to assert that *sig* is a valid signature of AP on (P, *r*).

   b) If *sig* is valid then U is successfully authenticated towards RP. Otherwise, authentication fails.

B.4.5 Analysis and UL classification

Protocol 3 enjoys the following properties:

a) Correctness: this is easily checked from the description of the protocol;

b) Unforgeability: observing the unforgeability game, it is easily seen that in order to pass the signature verification step 6) a), U needs to forge a signature on (P,*r*) under the signing key sk_{AP}. This would require breaking the underlying signature scheme;

   NOTE It can be proved formally that if the signature scheme is universally unforgeable, then the adversary fails with overwhelming probability.

c) PO-U unlinkability: since the messages exchanged at authentication time are either encrypted or do not depend on U, the PO-U adversary cannot win with probability significantly better than one half. Therefore Protocol 3 is PO-U unlinkable.

d) AO-U unlinkability: to find which of U_0 or U_1 is involved in the challenge authentication, the AO-U adversary faces an encryption under the private key of AP. The only way of attack consists in carrying out a man-in-the-middle attack to substitute the encryption key of AP at user registration time. However, to remain undetected by U, this attack would require the forgery of s_{AP}., which amounts to breaking the underlying signature scheme. Therefore, under standard security assumptions on the encryption and signature schemes used by AP, Protocol 3 is AO-U unlinkable.

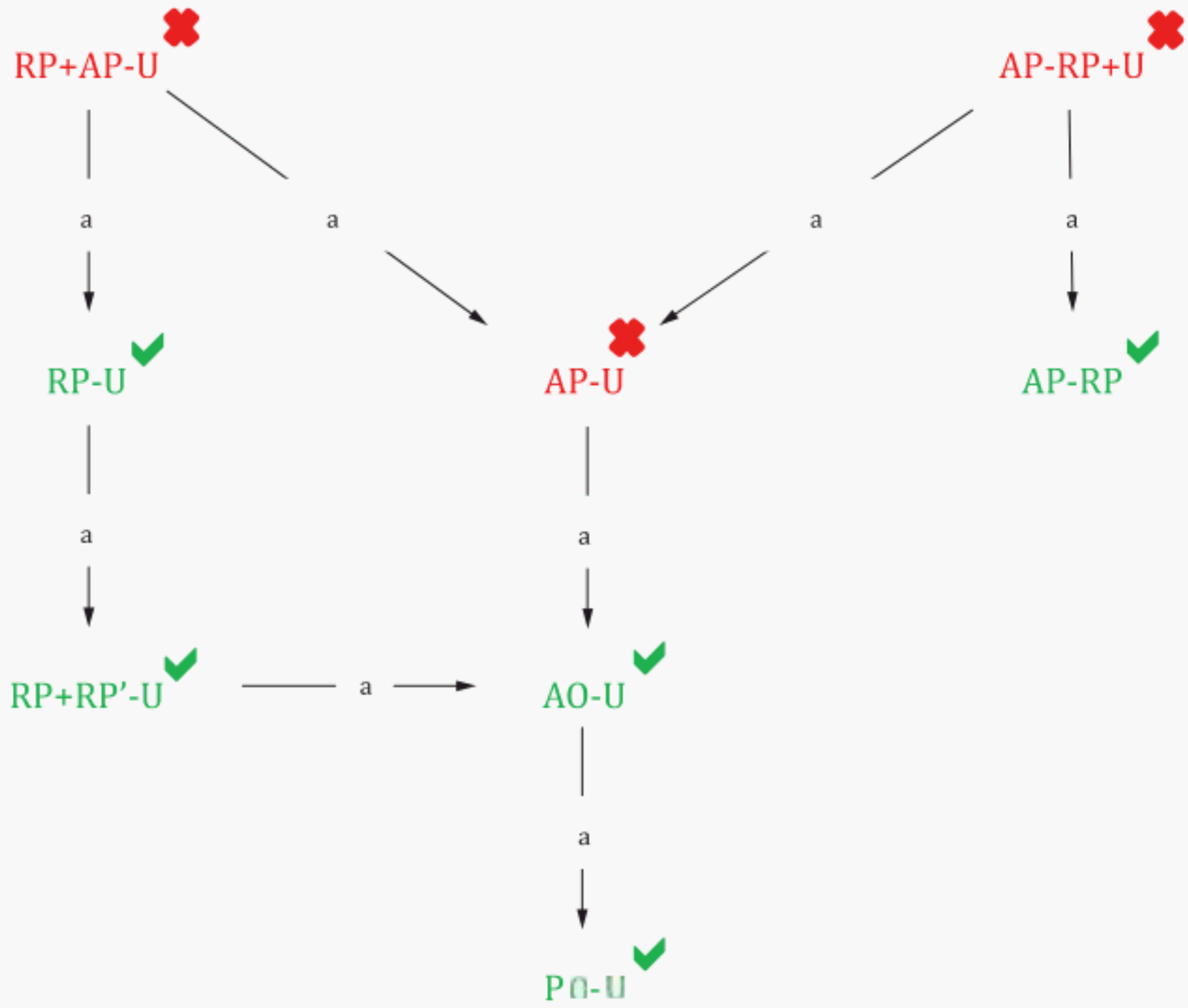
e) RP-U unlinkability: in addition to the power of active outsiders, the RP-U adversary also controls RP. However, RP does not have any private information that would help distinguish which of U_0 or U_1 is involved in the challenge authentication. Hence, Protocol 3 is RP-U unlinkable.

f) AP-RP unlinkability: since there is no information about RP in the contents of the messages exchanged at authentication time, RP remains unlinkable towards AP. Hence, Protocol 3 is AP-RP unlinkable.

When looking at the definition of AP-U unlinkability, the adversary can now access the private keys of AP. Therefore, A can decrypt the contents of the message sent by U_b and determine which of vk_{U0} or vk_{U1} appears in the plaintext. This allows A to determine *b* with probability one. Hence, Protocol 3 is not AP-U unlinkable.

Since Protocol 3 is not AP-U unlinkable, it cannot be either RP+AP-U or AP-RP+U unlinkable.

Our findings are summarized on ~~Figure B.3~~.



[a] Implies.

Figure B.3 — Unlinkability properties achieved by Protocol 3.

Applying ~~Table 2~~, Protocol 3 belongs to the class UL 3+.

Annex C
(informative)

C.1 General

This annex describes the following implementations of ABUEA:

1) OpenID Connect;

2) FIDO.

C.2 Implementing ABUEA with OpenID Connect self-issued OP

C.2.1 General

OpenID Connect, a popular identity federation protocol, can be used to provide an example of an attribute-based authentication protocol that achieves RP-U unlinkability. The protocol belongs to class UL 3+.

There are multiple ways of using OpenID Connect to achieve ABUEA. This subclause uses a self-issued OpenID provider (SIOP) to achieve it, hereafter called OpenID Connect Self-Issued ABUEA protocol. It is worth noting that even with the class of SIOP protocols, what is being explained is only one of the simplest example and there can be other ways to implement ABUEA.

In this model, the following actors are present:

— self-issued OpenID provider (SIOP) acting as U in ABUEA;

— claims provider acting as AP in ABUEA;

— client acting as RP in ABUEA;

SIOP resides on user's machine. It is assumed that user-agent is not identifiable via user-agent metadata in this protocol.

It is also assumed that the RP trusts the AP. The exact mechanism for the establishment of the trust is out of the scope of this document but can involve trust framework operator that provides assurance on the operation quality of the AP.

The protocol features preliminary phases (a setup phase and a user registration phase) that are carried out before the authentication phase itself.

C.2.2 Setup phase

1) AP generates an asymmetric key pair (sk_{AP} , vk_{AP}) for a digital signature mechanism.

2) AP generates an asymmetric key pair (ek_{AP} , dk_{AP}) for an encryption mechanism.

3) RP obtains the public verification key vk_{AP} of AP.

C.2.3 User registration phase

1) AP sends its public keys (vk_{AP}, ek_{AP}) to U as well as a signature s_{AP} on ek_{AP} under sk_{AP}.

2) U uses the verification key vk_{AP} of AP to verify that s_{AP} is a valid signature on ek_{AP}.

3) U generates an asymmetric key pair (sk_U, vk_U) for a digital signature mechanism.

4) U sends its public verification key vk_U to AP.

5) U and AP jointly agree on a set of attributes $A_U = \{a_1, ..., a_n\}$ that will be attached to U.

6) AP records (vk_U, A_U) in a database.

C.2.4 Authentication phase

1) U Generates an asymmetric key pair (sk_{UR}, vk_{UR}) towards the RP for a digital signature mechanism.

2) U initiates the transaction by sending a request including vk_{UR} to RP.

3) RP sends an attribute policy P (which is called *claims request* in OIDC) to U as well as a random number *r* (which is called *nonce* in OIDC) by invoking user-agent through "openid:" custom scheme.

4) U:

   a) Uses its signing key sk_U to generate a signature *s* on (P, *r*).

   b) Encrypts (P, *r*, *s*, vk_U) under the encryption key ek_{AP} of AP to obtain a ciphertext *c*. (Alternatively, U and AP can establish TLS connection to achieve the channel encryption, in which case, U authenticates the AP through the server certificate.)

   c) Sends *c* to AP.

5) AP:

   a) Uses its decryption key dk_{AP} to decrypt *c* and parses the resulting plaintext as (P, *r*, *s*, vk_U).

   b) Searches for a record (vk_U, A_U) in its database.

   c) Uses vk_U to verify that *s* is a valid signature of U on (P, *r*).

   d) Derives policy P_A from P that the AP believes to be equivalent to P and asserts that $P_A(A_U)$ = *true*.

   e) Uses its signing key sk_{AP} to generate a signature *sig* on (P_A, *r*).

   f) Sends *sig* to U. If any of the above steps fails for any reason, then AP aborts.

6) U:

   a) Uses its signing key sk_{UR} to generate a signature sig_u on *sig*.

   b) forwards vk_{UR}, sig_u and *sig* to RP over TLS as ID Token. (Optionally, ID Token can be encrypted by RP's encryption key.)

7) RP:

   a) Uses the verification key vk_{UR} of U to assert that sig_u is a valid signature of U on *sig*.

   b) Uses the verification key vk_{AP} of AP to assert that *sig* is a valid signature of AP on (PA, *r*).

   c) Compares P_A and P to establish the equivalence.

   d) If *sig* is valid and P_A and P is evaluated to be equivalent, then U is successfully authenticated towards RP. Otherwise the attribute authentication fails.

C.2.5 Analysis and UL classification

OpenID Connect self-issued ABUEA protocol enjoys the following properties:

a) correctness: this is easily checked from the description of the protocol;

b) unforgeability: observing the unforgeability game, it is easily seen that in order to pass the signature verification step 7) b), U needs to forge a signature on r under the signing key sk_{AP}. This would require breaking the underlying signature scheme;

NOTE It can be proved formally that if the signature scheme is universally unforgeable, then the adversary fails with overwhelming probability.

c) PO-U unlinkability: since the messages exchanged at authentication time are either encrypted or do not depend on U, the PO-U adversary cannot win with probability significantly better than one half. Therefore, this protocol is PO-U unlinkable;

d) AO-U unlinkability: to find which of U_0 or U_1 is involved in the challenge authentication, the AO-U adversary faces an encryption under the private key of AP. The only way of attack consists in carrying out a man-in-the-middle attack to substitute the encryption key of AP at user registration time. However, to remain undetected by U, this attack would require the forgery of s_{AP}., which amounts to breaking the underlying signature scheme. Therefore, under standard security assumptions on the encryption and signature schemes used by AP, this protocol is AO-U unlinkable;

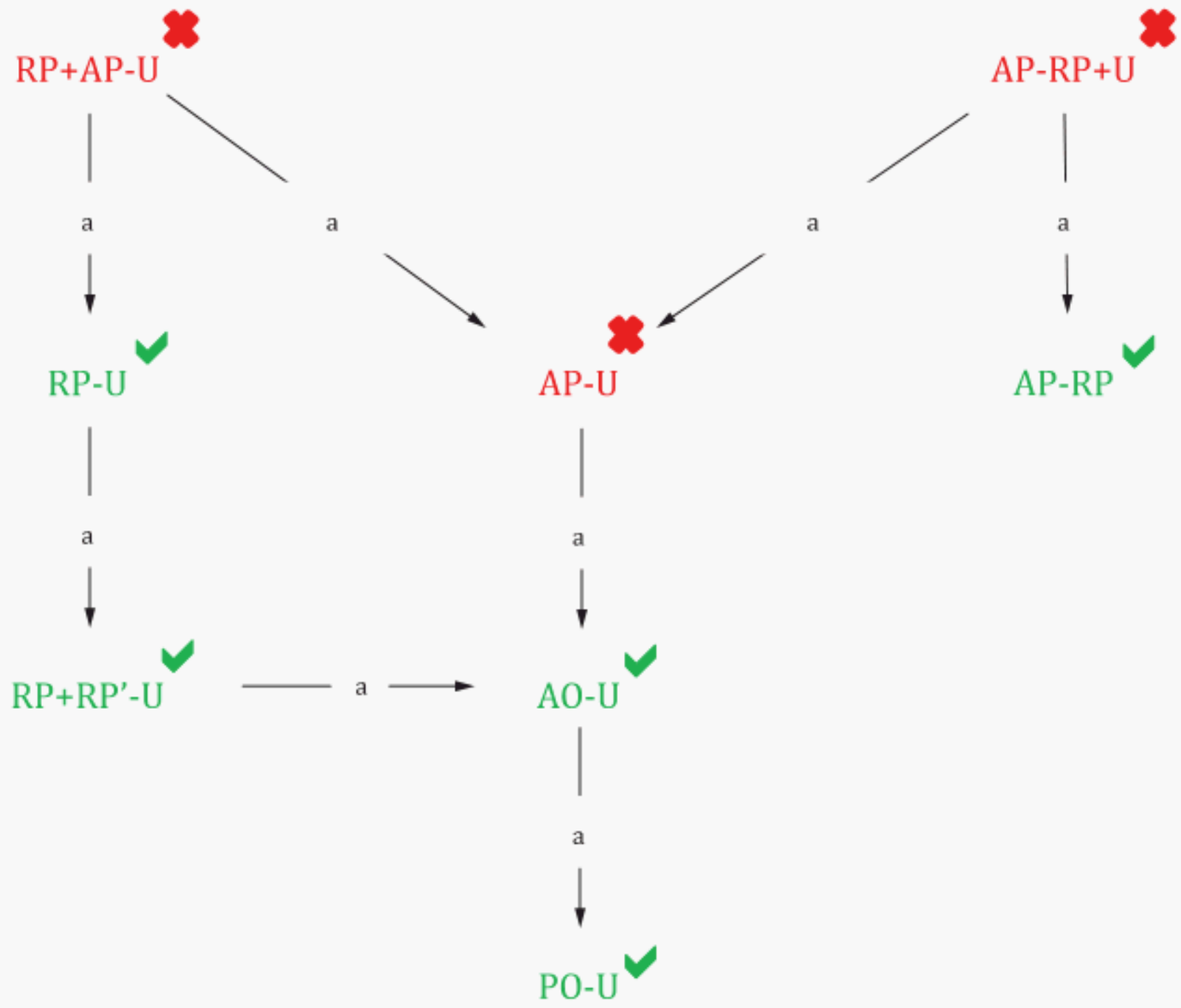
e) RP-U unlinkability: in addition to the power of active outsiders, the RP-U adversary also controls RP. However, RP does not have any private information that would help distinguish which of U_0 or U_1 is involved in the challenge authentication. Hence, this protocol is RP-U unlinkable;

f) AP-RP unlinkability: since there is no information about RP in the contents of the messages exchanged at authentication time, RP remains unlinkable towards AP. Hence this protocol is AP-RP unlinkable.

When looking at the AP-U game, it can be seen that the adversary can now access the private keys of AP. Therefore, A can decrypt the contents of the message sent by U_b and determine which of vk_{U0} or vk_{U1} appears in the plaintext. This allows A to determine b with probability one. Hence, this protocol is not AP-U unlinkable.

Since this OpenID Connect self-issued ABUEA protocol is not AP-U unlinkable, it cannot be either RP+AP-U or AP-RP+U unlinkable.

Our findings are summarized on ~~Figure C.1~~.



[a] Implies.

Figure C.1 — Unlinkability properties achieved by ABUEA with OpenID Connect self-issued

Applying ~~Table 2~~, this particular implementation of OpenID Connect belongs to the class UL 3A+ (RP-U and AP-RP)

C.3 Implementing ABUEA with FIDO

C.3.1 General

FIDO 2.0, a popular entity authentication protocol, can be used to provide an example of an attribute-based authentication protocol that achieves RP+RP'-U unlinkability. The protocol belongs to class UL 2A+ and UL3B+.

It is often mistaken that FIDO 2.0 does not give any attributes but this is not the case. Device attestation can be seen as providing certain attribute about the authenticator, e.g. who created the authenticator and what algorithm is supported.

In this model, the following actors are present:

— FIDO 2.0 authenticator acting as U in ABUEA;

— authenticator issuer acting as AP in ABUEA;

— relying party as RP in ABUEA;

FIDO authenticator performs communication through platform browser that resides on user's machine. It is assumed that the platform browser is not identifiable via user-agent metadata in this protocol.

It is also assumed that the RP trusts the AP. The exact mechanism for the establishment of the trust is out of scope of this document but can involve trust framework operator that provides assurance on the operation quality of the AP.

In this use case, policy P is whether or not the user has the authenticator issued by the AP.

The protocol features preliminary phases (a setup phase and a user registration phase) that are carried out before the authentication phase itself.

C.3.2 Setup phase

1) AP generates an asymmetric key pair (sk_{AP} , vk_{AP}) for a digital signature mechanism and embeds it in the authenticator.

2) RP obtains the public verification key vk_{AP} of AP.

C.3.3 User registration phase

1) AP directly or indirectly sends the authenticator, U, to the user.

C.3.4 Authentication phase

1) (This can be done once for each RP_i)
   a) RP_i sends its domain string $RPID_i$ to U together with an attribute policy P (which is an authentication request including RPID that requires certain attestation in FIDO 2.0).
   b) U generates an asymmetric key pair (sk_{Ui}, vk_{Ui}) for a digital signature mechanism for RP_i.
   c) U invokes an attestation function in itself that signs vk_{Ui} using sk_{AP} to create $sig0_i$.
   d) U sends vk_{Ui} and $sig0_i$ to RP_i.
   e) RP_i verifies *sig0* and stores vk_{Ui} .
2) RP sends U a random number *r* (which is called *nonce* in FIDO 2.0) by invoking U through WebAuthn browser API.
3) U (for RP_i):
   a) Uses its signing key sk_{Ui} to generate a signature sig_i on *r*.
   b) forwards vk_{Ui}, sig_i and *sig* to the system browser, which forwards the data as a response object to the RP via a TLS protected channel.
4) RP_i:
   a) Uses the verification key vk_{Ui} of U to assert that sig_i is a valid signature of U on *r*.
   b) Uses the verification key vk_{AP} of AP to assert that *sig* is a valid signature of AP on vk_{Ui}.
   c) If *sig* is valid then U is successfully authenticated towards RP_i. Otherwise the attribute authentication fails.

C.3.5 Analysis and UL classification

FIDO 2.0 ABUEA protocol enjoys the following properties:

a) correctness: this is easily checked from the description of the protocol;

b) unforgeability: observing the unforgeability game, it is easily seen that in order to pass the signature verification step 3) b), U needs to forge a signature on *r* under the signing key sk_{AP}. This would require breaking the underlying signature scheme;

NOTE It can be proved formally that if the signature scheme is universally unforgeable, then the adversary fails with overwhelming probability.

c) PO-U unlinkability: since the messages exchanged at authentication time are either encrypted or do not depend on U, the PO-U adversary cannot win with probability significantly better than one half. Therefore, this protocol is PO-U unlinkable.

d) AO-U unlinkability: to find which of U_0 or U_1 is involved in the challenge authentication, the AO-U adversary faces an encryption under the TLS of the channel between the system browser and the RP. The only way of attack consists in carrying out a man-in-the-middle attack to substitute the encryption key user by TLS between the system browser and the RP. However, to remain undetected by the system browser, this attack would require the forgery of *RPID,* which amounts to taking over RP, which now turns the issue at hand to RP-U unlinkability. Therefore, under standard security assumptions on the encryption and signature schemes used by AP, this protocol is AO-U unlinkable.

e) AP-U unlinkability: in addition to the power of active outsiders, the AP-U adversary also controls AP. However, since AP is not involved in the authentication, it cannot distinguish which of U_0 or U_1 is involved in the challenge authentication. Hence, this protocol is AP-U unlinkable.

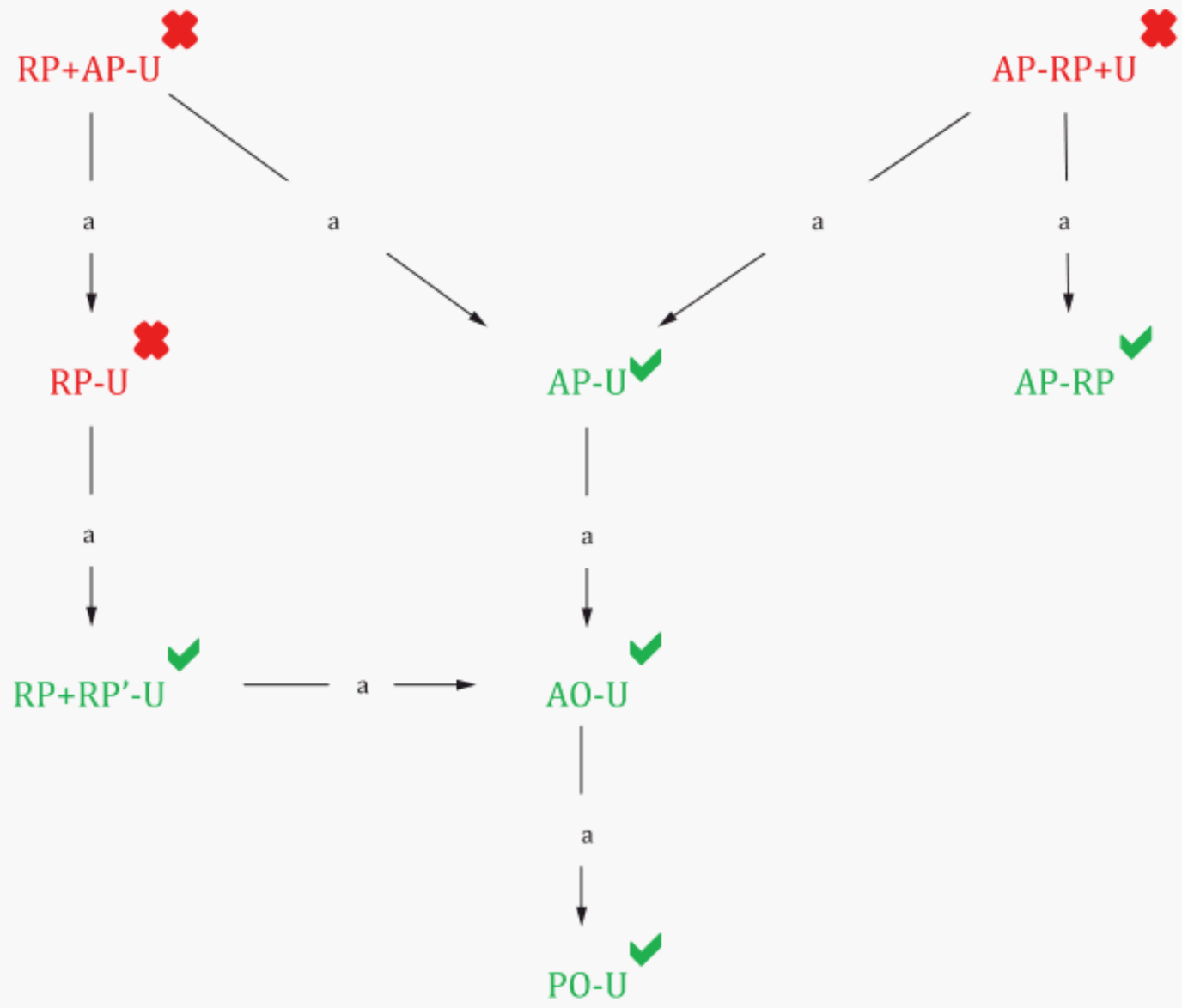
f) RP+RP'-U unlinkability: In FIDO 2.0, the key pairs to be used for the entity authentication is generated by RPID. Thus, even if RP and RP' colludes, it is not possible to correlate the run of the authentications. Thus, FIDO 2.0 is RP+RP'-U unlinkable.

g) AP-RP unlinkability: since there is no information about RP in the contents of the messages exchanged at authentication time, RP remains unlinkable towards AP. Hence this protocol is AP-RP unlinkable.

When looking at RP-U game, in addition to the power of active outsiders, the RP-U adversary also controls RP. In the case of FIDO 2.0, the user key pair is bound to the RP and the user transactions are linkable by RP, i.e. RP has private information that would deterministically distinguish which of U_0 or U_1 is involved in the challenge authentication. Hence, this protocol is not RP-U unlinkable.

Since this FIDO ABUEA protocol is not RP-U unlinkable, it can neither be RP+AP-U unlinkable nor AP-RP+U unlinkable.

Our findings are summarized on ~~Figure C.2~~.



[a] Implies.

Figure C.2 — Unlinkability properties achieved by ABUEA with FIDO 2.0

Applying ~~Table 2~~, this particular implementation of FIDO 2.0 belongs to the class UL 3B+ (AP-U and AP-RP)

Annex D
(informative)

Use cases for attribute-based unlinkable entity authentication

D.1 Overage verification

In many cases, only the fact that the user is over certain age is required to provide a service. For example, R-13 movie can only be provided to a user who is over 13 years old and above but no further.

D.2 Underage verification

In many cases, only the fact that the user is under certain age is required to provide a service. For example, a service that is providing safe place for children under certain age applies. It is like Overage verification, but there is one difference. Unlike overage verification which will be valid ever after, underage verification is only true for certain period of time.

Care should be taken as an expiry date can leak the birthday information of the PII principal.

D.3 Registered domicile verification

There are services that are only available to a resident of a certain district. In such a case, only the fact that the registered domicile address falls into the district is needed for the service provision.

Bibliography

[1] ISO/IEC/TS 29003:2018, *Information technology — Security techniques — Identity proofing*

[2] Tor Project ~~https://www.torproject.org~~

ICS 35.030

Price based on 34 pages