Guidelines for Secure Operation of Attribute Authorities and other issuers of access-granting statements

Publication Date	2018-11-22
Authors:	David Groep;David Kelsey;Hannah Short;Mischa Sallé;Uros Stevanovic;Stefan Paetow;Maarten Kremers
Document Code:	AARC-G048
DOI:	10.5281/zenodo.3234926
Grant Agreement No.:	730941
Work Package:	Policy and Best Practice Harmonisation
In collaboration with:	IGTF Interoperable Global Trust Federation

This guideline is a joint work of the International Global Trust Federation IGTF, the AARC project, and global partners. The research leading to these results has also received funding from the European Community's Horizon2020 Programme under Grant Agreement No. 730941 (AARC2).

Abstract

These guidelines describe the minimum requirements and recommendations for the secure operation of Attribute Authorities and similar services providing statements for the purpose of obtaining access to infrastructure services. Stated compliance with these guidelines may help to establish trust between issuers and Relying Parties. This document does not define an accreditation process.

Table of Contents

1. About this document

Associating properties to entities (be they persons, identities in general, or themselves groups or roles) may be done in a variety of different ways. Similarly, the conveyance of these properties, and their binding to entities, varies depending on the architectural model of the authentication and authorization system. Yet regardless of the model chosen, trust placed in the attributes relies on the operational security integrity of the authority that manages them. The guidance in this document concerns

- operational security processes and procedures for attribute authorities that ensure baseline information security practice
- requirements on traceability, auditability, and logging that ensure operational security events involving attribute authorities can be analysed and mitigated
- requirements on secure (integrity-protected and confidential) operation of the attribute authority
- requirements on secure (integrity-protected and confidential) interaction with the attribute authority

The latter two elements are partially dependent on the architectural model chosen for the authoritative attribute source. This document therefore distinguishes technology profiles for attribute authorities:

- attribute authorities that permit binding of properties to entities by means of lookup in which the entity whose properties are sought is the key in the look-up ('pull model')
- attribute authorities that issue (usually integrity-protected and, optionally, confidentiality-protected) statements in which attributes are asserted ('push model')

The storage and processing of re-usable credential data is outside the scope of this document, but should comply with relevant credential issuer guidance. Storage and processing of such data is strongly discouraged.

Examples of the 'pull model' include the use of an LDAP directory which is queried by one or more access control decision points, such as used in the PRACE Infrastructure, or a web service look-up such as in the OIDC UserInfo end-point, or the Grid User Management System (GUMS) as used by the Open Science Grid.

Examples of the 'push model' include issuing authorities that hand out signed attribute statements that can be presented to an access control system, such as used in the Virtual Organisation Membership Service VOMS by means of embeddable attribute certificates, the SAML assertions in a SAML2Int federation, or JWT tokens in an OAuth2 scenario.

Where guidance in this document are specific for one of these technology profiles, such requirements are indicated by being set in profile-specific text boxes. Such guidance then applies only to that technology profile. Guidance that is not so-indicated applies to all technology profiles, both pull- and push-based attribute authorities.

In this document the key words must, must not, required, shall, shall not, recommended, may, and optional are to be interpreted as described in RFC 2119. If a should or should not is not followed, the reasoning for this exception must be documented and published by the AA Operator such that relying parties can decide whether to accept the exception.

1.1. Definitions

Community

A set of one or more groups and sub-groups of persons (Users), organised with a common purpose and that are jointly granted access to one or more Infrastructures.

Community Management

A management body responsible for the Community and all its sub-groups, and for managing the lifecycle of user membership.

AA (Attribute Authority)

The technical entity operated by/for the Community to bind attributes, that may be used in authorization decisions, to subjects.

AA Operator

An organization, a group of organizations under single administrative control for the purpose of AA Operations, or group within an organization, that runs one or more Attribute Authorities.

Attribute

An attribute is a named property associated with an entity.

Subject

A subject is an entity, whose identity can be authenticated, and is a member of the Community.

Attribute Assertion

An attribute assertion is a statement, made by the Attribute Authority, about the attributes of a subject, and which may include time of issuance, may include time of expiration, and may be signed

Relying Party

The service or responsible entity that decides whom to trust, and estimates quality by evaluation of the statements it receives, and which consumes and processes attributes.

2. Introduction

These guidelines describe the minimum requirements and recommendations for the secure operation of Attribute Authorities, and similar services providing statements for the purpose of obtaining access to infrastructure services, by AA Operators.

In a typical scenario, a Community selects one or more AA Operators to operate AAs on their behalf, and informs relying parties of the related metadata. The attributes are securely maintained by the AA Operator and delivered on request to authorised relying parties. These attributes may be aggregated with identity assertions, delivered from a directory, a service API, or be in the form of attribute or capability tokens as asserted by an AARC BPA Proxy. The attributes may be used by relying parties for making authorisation decisions.

Stated compliance with these guidelines may help to establish trust between issuers and Relying Parties. In the interest of scalability, these guidelines are intended to facilitate the assessment of Attribute Authority Operators rather than individual Attribute Authorities or Communities. This document does not provide guidance on the management (life cycle, technical implementation, exchange protocols etc.) of attributes or the processes by which attributes are entered into the AA.

3. Operational Guidelines

3.1. Naming

There must be a way for a relying party to determine who issued or provided the attributes. Since attributes are used both for authorization decisions (in access control policies) and for identifying ownership of stateful resources (files in a storage system for instance), and the persistence of such state in relying party systems is outside the control of the AA or AA operator. Any re-assignment or relinquishing of name ownership will thus put user data and relying parties at risk. So names must not only be unique at any point in time, but also persistent.

1. Identifiers of the AA Operator and the AA must both be persistent and globally unique. Identifiers should be chosen in accordance with the AARC Guidelines and the Community Membership Management policy [AARC-G003]. The AA must use a defined naming scheme for subjects and attributes. Subject identifiers must be persistent and unique within an AA.

In case names are used whose semantics do not in themselves guarantee uniqueness (e.g. while URNs are guaranteed to be persistent, names in the domain name system could be re-assigned if they are inadvertently relinquished), the AA Operator and the AA must make a determined effort to impose such uniqueness. For example, domain name registrations must not be allowed to lapse, and a mechanism to ensure registration renewal in the future should be in place (e.g. by using subdomains under a long-lived organization, and not using project domain names unless due organizational care is taken).

Pull model

For attributes that are looked up by the relying party, the identifier the RP will have is the one from the AA operator: typically the directory service end-point URL. It is RECOMMENDED that the identifier persistency of the AA operator is also ensured through solely the domain name component of the URL. That is, the domain name used for the services by the AA operator must never be relinquished and remain tied to the AA operator.
 	
Good example:
`ldaps://ldap.surf.nl/ou=xenon,dc=co,dc=surf,dc=nl`
Bad example:
`ldap://ec2-54-247-51-111.eu-west-1.compute.amazonaws.com:3389/o=beauty-vo`

Push model
Issued assertions must contain the community (AA) name embedded in the assertion. Where possible, the community name should be chosen as AA identifier, and should be the same as the “scope” domain name component as used in the assignment to identifiers of community users and in the naming of community attributes in the “Guidelines and the Community Membership Management policy” [AARC-G003]. For AA operators that provide a multi-tenancy service for communities created ad-hoc, the naming may be based on an identifier assigned to the AA operator and then further specialized. Some assertion models have the ability to contain independently both the AA operator name (issuer) and the AA name (authoritative). For such systems, both elements must be unique and persistent. E.g. for the VOMS system, the AA operator name shall be the issuer of the attribute certificate (a directoryName assigned or delegated by an independent authority such as the IGTF, or ISO/ITU-T) and the attribute authority name based on the scope element as described in AARC-G003.

3.2. Attribute Management and Attribute Release

1. The Community must define and document the semantics, lifecycle, data protection, and release policy of attributes stored or asserted by the AA.

Follow the guidance from the Community Membership management policy recommendations; use standardised attributes as far as possible, and don’t deviate from the agreed semantics of such attributes;

Communities making modifications to the attribute set, its semantics, or release policies must proactively obtain agreement from the AA Operator for such changes.

2. The AA Operator must implement the community definitions as defined and documented, for all the AAs it operates.

By implementing these requirements, the AA operator will support the chain of trust between Community and the RPs.

An AA Operator must only host those communities for which it can implement the requirements.

3. The AA Operator must collect and publish the community documents for the benefit of Relying Parties.

The AA Operator is typically the only interaction point that the RPs will know about, and thus the only way for them to gain insight into the Community policies and practices and thus evaluate the level of reliance to place in the attributes obtained.

3.3. Attribute Assertions

1. Assertions provided by an AA must be integrity-protected. They must be signed by the identified AA, or be transmitted over an integrity-protected channel where the server has been authenticated, and preferably both.

Push model
Where the protocol supports it, enable protection also of the messages conveyed over the established channel. Good examples: SAML Attribute Query should enable message signing and use TLS.

Pull model
As a good example: LDAP should enable TLS protection of the channel

2. The AA must respect data protection requirements of the Infrastructure and Community. This may mean that AAs require client authentication, in addition to the encryption of the messages and the communication channel.

Pull model
In the OIDC dynamic client registration model, the protection put in place to prevent the client from retrieving arbitrary data from the OP is the introduction of an interstitial user authentication point and asking agreement for attribute release from the user. This constitutes an implementation of ‘client authentication’ as intended in the clause above. In models where a trusted federation is in place, this may be replaced by this third-party trust model within the federation, depending on the specific use case and data protection requirements.

3. If an AA Operator issues assertions containing a lifetime, this lifetime must be compliant with the Community policies, be no more than 24 hours, and the assertion must not be valid beyond the validity period of the attributes it contains. The Community Management is responsible for the content of the assertion, as issued, during its entire lifetime.

These guidelines do not require a revocation mechanism for issued attribute assertions.

4. Re-issuance of assertions must be based on information held in the AA at the time of re-issuance.

OIDC refresh tokens are re-presented to and validated by the issuer and at that point they can either issue an access token or not. Refresh tokens are conceptually equivalent to user credentials and, as such, attribute issuers should follow guidance for identity issuing authorities.

5. The AA Operator must only issue assertions or release attributes to requesters in accordance with the Community policies.

3.4. Operational requirements

1. An AA that issues attribute assertions must be a dedicated system, running no other services than those needed for the AA operations.
2. An AA may be run in a virtual environment that has security requirements the same or better than required for the AA, and for all services running in this environment, and it must not leave this security context. Any virtualization techniques employed (including the hosting environment) must not degrade the context as compared to any secured physical setup. Only AA Operator designated personnel should have control over the virtualisation and security context of the AA.
3. The AA must be located in a secure environment where access is controlled and limited to specific trained personnel.
4. The AA must be run with an intended continuous availability. Hosted Communities must be informed if AA Operator procedures change.
5. To achieve sustainability, an AA Operator should offer its AA services as a long-term commitment.

3.4.1. Key Management

1. A key used to protect assertions should be dedicated to assertion protection functions.

Push model
If the AA both signs assertions and provides functionality over protected channels, the keys used to sign assertions shall be different from those protecting those channels.

Pull model
The key of the AA must be used solely for protecting connections to its protocol endpoint and ensure an integrity protected and mutually authenticated channel.

2. Keys must not be shared between AA Operators. A single AA Operator may use the same signing key for multiple AAs. Where multiple AAs are under the control of a single AA operator but located in physically distributed locations, the key must only be shared using secure protocols.

For example, an AA Operator can be physically distributed but still constitute a single 'organisation' if there is a common managerial control over all distributed instances. It must then have a single PMA and a single operations coordination team, which is under single management (of the PMA). This is one operator. Key distribution must in this case be controlled by a protocol that protects against compromise in transit and at rest. Where a community commissions multiple independent service providers to run their AA, and even if it is the same namespace the operators are not under common management, and no binding organisational measures have been agreed to ensure all operators to the same things. So they must use different keys, so an RP can choose to trust one and not the other.

3. Keys must have a protection strength equivalent to 112 bits (symmetric) or higher.

To compare implementations, refer to e.g. Special Publication 800-57 Part 1 Rev 4 NIST, 01/2016. 112-bit symmetric encryption is equivalent to 2048-bit RSA, 128-bit encryption to 3072-bit RSA or 256- to 383-bit ECC. Anything above that (e.g. 192 symmetric bits) is hard to express in workable RSA keys (8192-bit and larger), and should preferably be protected through ECC.

4. Keys must only be accessible by the service and by trained personnel subject to procedural controls.

Pull model
The keys referred to above are those used to authenticate and integrity protect the channel.

5. AA Operators are encouraged to consider using an HSM to store signing keys. Otherwise, when using software-based private keys these must be suitably protected by the operating system.

When keys are not stored in an HSM, they should preferably exist in memory only (and key daemon mechanisms may be employed in high-availability environments to maintain service continuity by key re-synchronisation). Only in exceptional cases, and only with supplementary controls, should keys be held on persistent storage and be protected solely by storage and file system level permissions.

3.4.2. Network Configuration

6. The network to which the AA system is connected must be highly protected and suitably monitored.

Service access should be protected by at least two distinct control layers not running the same software or operating system, and the AA system must not run any unnecessary services. The network should be monitored for anomalous events, such as detection of data exfiltration, credential probing, and brute-force attacks. It should preferably also be protected

against denial of service attacks in order to prevent security downgrade attacks and unexpected induced fail-overs.

3.5. Site Security

1. The AA Operator should document the physical site security controls and maintain them in a state consistent with the security requirements of the hosted Communities.

For example, the system should be in a locked room with auditable physical access controls, protected against intrusions and be at least tamper-evident in case of such intrusions.

Additional elements such as flood and fire protection, or protection against threats emerging as a result of providing service to specific communities, should be considered as well.

3.6. Metadata publication

1. The AA Operator must publish at least the following metadata for each AA it hosts, to the Community and related relying parties:
   a) administrative contact details for the AA Operator, including at least one email address and one postal contact address
   b) an operational security contact for the AA Operator, being at least an email address and preferably including a telephone number,
   c) those aspects of their operational environment that are relevant to the evaluation of the security and trust by the Communities and Relying Parties
   d) the public key for verifying signed messages, where relevant, or the set of certificates up to a self-signed root;
   e) a web URL to a general information page about the Community

The operational security contact is expected to respond in a manner consistent with the Sirtfi requirements.

2. The AA Operator should provide a means to validate the integrity of its roots of trust.

For example by having it included in the meta-data feed of a trusted SAML or OIDC federation, by submitting it to a trust anchor repository such as TACAR, the IGTF trust anchor distribution, the trust anchor distribution of a pertinent e-Infrastructure, or the set of public web trust roots.

3.7. Assessments and auditability

1. The attributes in the AA and their binding to subjects must be verifiable and auditable.

The AA should be run based on a structured source of data, and the data source and its contents must be auditable.

2. The AA Operator must record and archive at least the following for all of its hosted AAs:
   a) all requests for attributes
   b) all issued attribute assertions
   c) any configuration change to the AA relevant to the access control of the attribute repository
   d) any change affecting the binding between subjects and attributes

3. The AA Operator must record and archive at least the following for of its AA issuance systems:
   a) all login/logout/reboot/key activations of the issuing system
   b) changes to the configuration of the issuing system

4. The AA Operator must keep these records after termination of the effects of the auditable event for as long as required by the Community and any relying parties that have entered into an agreement with the AA Operator, and as required by applicable legislation.

In absence of guidance, the recommended retention period is 400 days.

5. The AA Operator must provide assistance to operational security teams during a security incident.
6. The AA Operator must accept being audited following reasonable requests from a Community it serves and from relying parties that have entered into an agreement with the AA Operator, to verify its compliance with these guidelines.
7. The AA Operator should perform operational audits of its staff at least once per year. A list of AA Operator staff should be maintained, and verified at least once per year.

3.8. Privacy and confidentiality

1. AA Operators must define and publish an appropriate privacy and data release policy compliant with the relevant legislation and the requirements of the Community.

3.9. Compromise and disaster recovery

1. The AA Operator must have an adequate compromise and disaster recovery procedure, and must be willing to disclose this to the hosted Communities or to either an assessor or all related relying parties.

4. Relying Party obligations

1. If a Community uses AAs operated by multiple AA Operators then Relying Parties must assess each of the AA Operators individually.

2. Relying Parties must verify the integrity and validity of attribute assertions and any binding to a valid subject at the time of reliance.

If the time of reliance is shifted from the time of receipt of the assertion, e.g. in case of batch processing, the relying party should still verify the validity of any expired assertions by refreshing these at the time actions are taken.

3. Relying Parties must rely on assertions with an explicit lifetime only for as long as they are valid.
4. Relying Parties must assess the risk of relying on assertions with no explicit lifetime and should not rely on them for longer than 24 hours after issuance[1].

5. Relying Party must validate all verifiable elements[2].

[1] neither the AA Operator nor the AA are responsible for decisions based on information without a specified lifetime after the AA has updated its own database

[2] verifiable elements include such as the full certification path of the subject identity certificate, public keys used for signing assertions, etc.