Specification for expressing resource capabilities

Publication Date	2018-12-10
Authors:	AARC Consortium Partners;AppInt members;Marcus Hardt (ed.)
Document Code:	AARC-G027
DOI:	https://doi.org/10.5281/zenodo.2247446
Grant Agreement No.:	730941
Work Package:	JRA1

The research leading to these results has received funding from the European Community's Horizon2020 Programme under Grant Agreement No. 730941 (AARC2).

Abstract

This document provides *a specification for expressing resource-specific capabilities using entitlements. A capability defines the resource or child-resource a user is allowed to access, optionally specifying certain actions the user is entitled to perform. Capabilities can be used to convey - in a compact form - authorisation information.*

Table of Contents

1. Introduction

This document provides a specification for expressing resource-specific capabilities using entitlements. In the rest of this document, resource-specific capabilities will be referred to as just capabilities. A capability defines the resource or child-resource a user is allowed to access, optionally specifying certain actions the user is entitled to perform. Capabilities can be used to convey - in a compact form - authorisation information.

1.1. Conventions

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. Expression of capabilities

Capabilities SHOULD be expressed according to the following syntax (components enclosed in square brackets are OPTIONAL, three dots (‘`...`’) indicate additional entries of the type after which they are placed, the backslash (‘`\`’) being the continuation character):

```
<NAMESPACE>:res:<RESOURCE>[:<CHILD-RESOURCE>]...[:act:<ACTION>[,<ACTION>]...]#<AUTHORITY>
```

where:

- **<NAMESPACE>**[1] is controlled by the e-infrastructure, research infrastructure or research collaboration that manages the capability. It is in the form of

[1] The **<NAMESPACE>** definition follows that in [AARC-G002]

urn**:<NID>:<DELEGATED-NAMESPACE>[:<SUBNAMESPACE>]...**

where

- **<NID>** is the namespace identifier associated with a URN namespace registered with IANA[2], ensuring global uniqueness. Implementers SHOULD use one of the existing registered URN namespaces, such as urn:geant[GEANT] or urn:mace[MACE].
- **<DELEGATED-NAMESPACE>** is a URN sub-namespace delegated from one of the IANA registered NIDs to an organisation representing the e-infrastructure, research infrastructure or research collaboration. It is RECOMMENDED that a publicly accessible URN value registry for each delegated namespace be provided.

A **<NAMESPACE>** can have a variable number of elements. For example

- **urn:geant:edugain**
- **urn:geant:nikhef.nl**
- **urn:geant:nikhef.nl:idm**

are all valid **<NAMESPACE>** values.

- The literal string "**res**" indicates that this is a *resource-specific* entitlement as opposed to, for example, an entitlement used for expressing group membership [AARC-G002].
- **<RESOURCE>** is the name of the resource. Whether the name should be unique is an implementation decision.
- An optional list of colon-separated **<CHILD-RESOURCE>** components represents a specific branch of the hierarchy of resources under the identified **<RESOURCE>.**
- An optional list of comma-separated **<ACTION>**s MAY be included, which, if present, MUST be prefixed with the literal string "**act**". This component MAY be used for further specifying the actions a user is entitled to do at a given resource. Note that the list of **<ACTION>**s is scoped to the rightmost child-resource; if no child-resource information is specified, actions apply to the top level resource. The interpretation of a capability without actions specified is an implementation detail.
- **<AUTHORITY>** is a mandatory and non-empty string that indicates the authoritative source of the capability. This SHOULD be used to further specify the exact issuing instance. For example, it MAY be the FQDN of the service that issued that specific capability. The **<AUTHORITY>** is specified in the f-component [RFC8141] of the URN; thus, it is introduced by the number sign ("**#**") character and terminated by the end of the URN. All characters must be encoded according to [RFC8141]. Hence, the **<AUTHORITY>** MUST NOT be considered when determining equivalence (Section 3 in [RFC8141]) of URN-formatted capabilities.

[2] Generic top level namespaces require IANA approval as per Section 6.2 of [RFC8141]

References

[AARC-G002] AARC Recommendation on Expressing group membership and role information
https://aarc-project.eu/guidelines/aarc-g002

[GEANT] GEANT URN registry
https://www.geant.org/Services/Trust_identity_and_security/Pages/NamespaceRegistry.aspx

[MACE] Mace URN registries
https://www.internet2.edu/products-services/trust-identity/mace-registries/urnmace-namespace

[RFC2119] Key words for use in RFCs to indicate Requirement levels
https://tools.ietf.org/html/rfc2119

[RFC2141] URN Syntax
https://tools.ietf.org/html/rfc2141

[RFC6963] A Uniform Resource Name (URN) Namespace for Examples
https://tools.ietf.org/html/rfc6963

[RFC8141] Uniform Resource Names
https://tools.ietf.org/html/rfc8141

Appendix A: Examples

The examples provided below use the **urn:example** namespace [RFC6963]. The delegated namespace example-ri.org has been registered by the example research infrastructure.

A.1 Simple example without specified actions

A capability that denotes eligibility to access a specific resource named example-res.org with no specific permitted actions, could be expressed as follows:

```
urn:example:example-ri.org:res:example-res.org#auth-x.example-infra-b.org
```

A.2 Example with child-resource and multiple actions

A more complex example, allowing the actions create and delete of VM images in the child-resource **storage** of a resource called **vm_dashboard:**

```
urn:example:example-ri.org:\
res:vm_dashboard:storage:act:create,delete\
#auth-x.example-ri.org
```

A.3 Advanced examples

There may be cases in which more complicated authorisation statements need to be conveyed. The following example uses the concept of named queries to express an SQL statement:

```
urn:example:example-ri.org:res:database:act:SQL_USER_LIST_STATEMENT\
#auth-x.example-infra.org
```

This concept may be extended to other capabilities. For example:

```
urn:example:example-ri.org\
:res:se_52:act:read%3A%2Fstore,upload%3A%2Fstore%2Fmc%2Fdataset_a\
#auth-x.example-infra.org
```

Note that

```
read%3A%2Fstore,upload%3A%2Fstore%2Fmc%2Fdataset_a
```

is the result of individually encoding the actions **read:/store** and **upload:/store/mc/dataset_a** according to the rules defined in Section 2.2 of [RFC2141].