Internet Engineering Task Force (IETF)
Request for Comments: 6075
Updates: 2244
Category: Standards Track
ISSN: 2070-1721

D. Cridland
Isode Limited

December 2010

The Internet Assigned Number Authority (IANA) Application Configuration Access Protocol (ACAP) Vendor Subtrees Registry

Abstract

The original Application Configuration Access Protocol (ACAP) specification included a vendor registry now used in other protocols. This document updates the description of this registry, removing the need for a direct normative reference to ACAP and removing ambiguity.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc6075.

Table of Contents

1. Introduction

The [ACAP] specification includes the specification and creation of the ACAP Vendor Registry, and this registry has subsequently been reused by several specifications, including both [ANNOTATE] and [METADATA], and is proving to be a useful mechanism for namespacing various names to within a specific vendor's scope.

The use of textual rather than numeric identifiers for vendors benefits engineers and operators who are diagnosing protocol problems by allowing them some possibility of identifying the origin of a vendor attribute without having to look it up in a registry (although that remains a necessary fallback). As such, engineers and operators already have to be familiar with international technical English to diagnose textual protocol problems, the restriction to ASCII may help and is not believed to harm that intended use. Exposure of vendor attributes directly in end-user user interfaces was not an intended use of the registry.

This document merely updates the registry to reduce ambiguity in the original specification and dissociates it from the original document in all but name, allowing easier referencing. It replaces Section 7.4 and portions of Section 4, particularly Section 4.3, of [ACAP].

2. Conventions Used in This Document

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 [KEYWORDS].

The formal syntax is to be considered normative and is specified using [ABNF]. Where a formal syntax and the prose are in conflict, the formal syntax takes precedence.

3. The Vendor Subtree Registry

A Vendor Token is a UTF-8 string that begins with "vendor." and that is followed by the name of the company or product. This name MUST NOT contain any slash character, period, or the percent and asterisk characters typically used as wildcards.

Following this may be names, separated from the Vendor Token by a period, which need not be registered, thus forming a complete Vendor Name.

3.1. Internationalization

Vendor Tokens are able to contain any valid Unicode codepoint, encoded as [UTF-8], except the special characters. Since the publication of [ACAP], however, concerns have been raised on the handling and comparison of full Unicode strings; therefore, this specification restricts the current registrations to the ASCII subset of UTF-8.

Furthermore, characters such as ASCII control characters, most whitespace, and quotes are likely to be confusing and have been similarly restricted.

Therefore, this document allows only ASCII letters, digits, the hyphen, and space to be used in registrations (the <iana-vendor-tag> ABNF production in Section 3.2).

At the time of publication of this document, no existing registrations violate the new restricted syntax on characters allowed in registrations. [ACAP] required all Vendor Tokens to be registered with IANA, so the new restriction is not believed to introduce any interoperability issue.

Finally, note that this document does not change the requirement on processors to accept other non-ASCII Unicode codepoints in Vendor Tokens (the <possible-vendor-tag> ABNF production in Section 3.2).

3.2. Formal Syntax

This syntax draws upon productions found within [ABNF] and [UTF-8]. Productions replace those in Section 4.3 of [ACAP].

```
vendor-name         = vendor-token *("." name-component)

name-component      = *(name-char / UTF8-2 / UTF8-3 / UTF8-4)

name-char           = %x01-24 / %x26-29 / %x2B-2D / %x30-7F
                    ;; ASCII-range characters not including ".",
                    ;; "/", "%", or "*".

vendor-token        = "vendor." vendor-tag
                    ;; MUST be registered with IANA

vendor-tag          = iana-vendor-tag / possible-vendor-tag

iana-vendor-tag     = 1*(ALPHA / DIGIT / SP / "-")
                    ;; This production represents
                    ;; allowed forms for registrations
                    ;; under the rules specified in this
                    ;; document.

possible-vendor-tag = name-component
                    ;; This production represents what
                    ;; applications and specifications
                    ;; MUST be able to accept.
```

3.3. Examples

A company Example, Ltd. might register the Subtree "vendor.example". This means it may use "vendor.example", or any name at all beginning "vendor.example.", such as "vendor.example.product".

These names might be used in several protocols, and are reserved in all the relevant protocols, so "vendor.example" might be an ACAP [ACAP] dataset class name, and "/vendor/vendor.example" might be a tree of IMAP ANNOTATE entries [ANNOTATE].

Example, Ltd. is free to use either "vendor.example", and group specific products under it using the relevant protocol’s hierarchy -- perhaps "/shared/vendor/vendor.example/product" annotation [ANNOTATE], or using more specific names, such as "/shared/vendor/vendor.example.product" annotation.

Note that the solidus ("/") characters in the examples above are protocol delimiters that are themselves not part of the Vendor Token.

3.4. Changes from RFC 2244

This non-normative section details changes from the original specification of the registry in RFC 2244.

- Vendor Tokens are restricted to ASCII for registration purposes.
- Clarifications that "vendor.<company/product name>" means "vendor.company name" or "vendor.product name" - "vendor.company/product" is and always has been illegal.
- Made "vendor.company" a name in its own right - RFC 2244 only refers to a prefix of "vendor.company.".
- Added example registration, in line with [EXAMPLES].

4. IANA Considerations

This specification updates the IANA registry named the ACAP "Vendor Subtrees" registry. IANA has updated the registry to point at this document.

Vendors may reserve a portion of the ACAP namespace, which is also used as the namespace for several other protocols, for private use. Vendor Names are reserved for use by that company or product, wherever used, once registered. Registration is on a first come, first served basis. Whenever possible, private attributes and classes should be eschewed in favour of improving interoperable protocols.

Vendors may only use names conforming to iana-vendor-tag at the current time; future revisions of this specification may change this.

To: iana@iana.org
Subject: Registration of ACAP Vendor Subtree

Private Prefix: vendor.name

Person and email address to contact for further information:

(company names and addresses should be included where appropriate)

4.1. Example Registration

IANA is requested to add the following registration, for use by specification authors in examples, similarly to the domains specified in [EXAMPLES]:

To: iana@iana.org
Subject: Registration of ACAP Vendor Subtree

Private Prefix: vendor.example

Person and email address to contact for further information:

Dave Cridland <dave.cridland@isode.com>

5. Security Considerations

There are no known security issues with this registry. Individual protocols using Vendor Subtree names may have security issues, and the introduction of Unicode has, in itself, security implications -- the restriction of this is thought to mitigate these.

6. Acknowledgements

Thanks must go to Chris Newman, John Myers, and the other designers of ACAP for the initial creation of the registry. Thanks also to Alexey Melnikov for advice on this revision.

7. References

7.1. Normative References

[ABNF] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.

[ACAP] Newman, C. and J. Myers, "ACAP -- Application Configuration Access Protocol", RFC 2244, November 1997.

[KEYWORDS] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[UTF-8] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, November 2003.

7.2. Informative References

[ANNOTATE]
Daboo, C. and R. Gellens, "Internet Message Access Protocol - ANNOTATE Extension", RFC 5257, June 2008.

[EXAMPLES]
Eastlake 3rd, D. and A. Panitz, "Reserved Top Level DNS Names", BCP 32, RFC 2606, June 1999.

[METADATA]
Daboo, C., "The IMAP METADATA Extension", RFC 5464, February 2009.

Author's Address

Dave Cridland
Isode Limited
5 Castle Business Village
36, Station Road
Hampton, Middlesex TW12 2BX
GB

EMail: dave.cridland@isode.com