ACME Working Group                                             H. Landau
Internet-Draft                                          January 15, 2019
Intended status: Standards Track
Expires: July 19, 2019

# CAA Record Extensions for Account URI and ACME Method Binding
## draft-ietf-acme-caa-06

## Abstract

The CAA DNS record allows a domain to communicate issuance policy to CAs, but only allows a domain to define policy with CA-level granularity. However, the CAA specification also provides facilities for extension to admit more granular, CA-specific policy. This specification defines two such parameters, one allowing specific accounts of a CA to be identified by URI and one allowing specific methods of domain control validation as defined by the ACME protocol to be required.

## Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on July 19, 2019.

## Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction . . . . . . . . . . . . . . . . . . . . . . . . 2
2. Terminology . . . . . . . . . . . . . . . . . . . . . . . . 2
3. Extensions to the CAA Record: accounturi Parameter . . . . . 3
  3.1. Use with ACME . . . . . . . . . . . . . . . . . . . . . . 3
  3.2. Use without ACME . . . . . . . . . . . . . . . . . . . . 4
4. Extensions to the CAA Record: validationmethods Parameter . . 4
5. Security Considerations . . . . . . . . . . . . . . . . . . . 4
  5.1. Limited to CAs Processing CAA Records . . . . . . . . . . 4
  5.2. Restrictions Ineffective without CA Recognition . . . . . 5
  5.3. Mandatory Consistency in CA Recognition . . . . . . . . . 5
  5.4. URI Ambiguity . . . . . . . . . . . . . . . . . . . . . . 6
  5.5. Authorization Freshness . . . . . . . . . . . . . . . . . 6
  5.6. Use with and without DNSSEC . . . . . . . . . . . . . . . 7
  5.7. Restrictions Supercedable by DNS Delegation . . . . . . . 8
  5.8. Misconfiguration Hazards . . . . . . . . . . . . . . . . 8
6. IANA Considerations . . . . . . . . . . . . . . . . . . . . . 8
7. Normative References . . . . . . . . . . . . . . . . . . . . 8
Appendix A. Examples . . . . . . . . . . . . . . . . . . . . . . 9
Author's Address . . . . . . . . . . . . . . . . . . . . . . . . 9

## 1. Introduction

This specification defines two parameters for the "issue" and "issuewild" properties of the Certification Authority Authorization (CAA) DNS resource record [RFC6844]. The first, "accounturi", allows authorization conferred by a CAA policy to be restricted to specific accounts of a CA, which are identified by URIs. The second, "validationmethods", allows the set of validation methods supported by a CA to validate domain control to be limited to a subset of the full set of methods which it supports.

## 2. Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in BCP 14, RFC 2119 [RFC2119] and indicate requirement levels for compliant ACME-CAA implementations.

## 3. Extensions to the CAA Record: accounturi Parameter

A CAA parameter "accounturi" is defined for the "issue" and "issuewild" properties defined by [RFC6844]. The value of this parameter, if specified, MUST be a URI [RFC3986] identifying a specific CA account.

"CA account" means an object maintained by a specific CA representing a specific entity, or group of related entities, which may request the issuance of certificates.

The presence of this parameter constrains the property to which it is attached. Where a CAA property has an "accounturi" parameter, a CA MUST only consider that property to authorize issuance in the context of a given certificate issuance request if the CA recognises the URI specified as identifying the account making that request.

If a CA finds multiple CAA records pertaining to it (i.e., having property "issue" or "issuewild" as applicable and a domain that the CA recognises as its own) with different "accounturi" parameters, the CA MUST consider the CAA record set to authorize issuance if and only if at least one of the specified account URIs identifies the account of the CA by which issuance is requested.

A property without an "accounturi" parameter matches any account. A property with an invalid or unrecognised "accounturi" parameter is unsatisfiable. A property with multiple "accounturi" parameters is unsatisfiable.

The presence of an "accounturi" parameter does not replace or supercede the need to validate the domain name specified in an "issue" or "issuewild" record in the manner described in the CAA specification. CAs MUST still perform such validation. For example, a CAA "issue" property which specifies a domain name belonging to CA A and an "accounturi" parameter identifying an account at CA B is unsatisfiable.

### 3.1. Use with ACME

An ACME [I-D.ietf-acme-acme] account object MAY be identified by setting the "accounturi" parameter to the URI of the ACME account object.

Implementations of this specification which also implement ACME MUST recognise such URIs.

### 3.2. Use without ACME

The "accounturi" specification provides a general mechanism to identify entities which may request certificate issuance via URIs. The use of specific kinds of URI may be specified in future RFCs, and CAs not implementing ACME MAY assign and recognise their own URIs arbitrarily.

## 4. Extensions to the CAA Record: validationmethods Parameter

A CAA parameter "validationmethods" is also defined for the "issue" and "issuewild" properties. The value of this parameter, if specified, MUST be a comma-separated string of challenge method names. Each challenge method name MUST be either an ACME challenge method name or a CA-assigned non-ACME challenge method name.

The presence of this parameter constrains the property to which it is attached. A CA MUST only consider a property with the "validationmethods" parameter to authorize issuance where the name of the challenge method being used is one of the names listed in the comma-separated list.

Where a CA supports both the "validationmethods" parameter and one or more non-ACME challenge methods, it MUST assign identifiers to those methods. If appropriate non-ACME identifiers are not present in the ACME Validation Methods IANA registry, the CA MUST use identifiers beginning with the string "ca-", which are defined to have CA-specific meaning.

## 5. Security Considerations

This specification describes an extension to the CAA record specification increasing the granularity at which CAA policy can be expressed. This allows the set of entities capable of successfully requesting issuance of certificates for a given domain to be restricted beyond that which would otherwise be possible, while still allowing issuance for specific accounts of a CA. This improves the security of issuance for domains which choose to employ it, when combined with a CA which implements this specification.

### 5.1. Limited to CAs Processing CAA Records

All of the security considerations of the CAA specification are inherited by this document. This specification merely enables a domain with an existing relationship with a CA to further constrain that CA in its issuance practices, where that CA implements this specification. In particular, it provides no additional security above that provided by use of the unextended CAA specification alone

as concerns matters relating to any other CA. The capacity of any other CA to issue certificates for the given domain is completely unchanged.

As such, a domain which via CAA records authorizes only CAs adopting this specification, and which constrains its policy by means of this specification, remains vulnerable to unauthorized issuance by CAs which do not honour CAA records, or which honour them only on an advisory basis. Where a domain uses DNSSEC, it also remains vulnerable to CAs which honour CAA records but which do not validate CAA records by means of a trusted DNSSEC-validating resolver.

### 5.2. Restrictions Ineffective without CA Recognition

The CAA parameters specified in this specification rely on their being recognised by the CA named by an "issue" or "issuewild" CAA property. As such, the parameters are not an effective means of control over issuance unless a CA's support for the parameters is established beforehand.

CAs which implement this specification SHOULD make available documentation indicating as such, including explicit statements as to which parameters are supported. Domains configuring CAA records for a CA MUST NOT assume that the restrictions implied by the "accounturi" and "validationmethods" parameters are effective in the absence of explicit indication as such from that CA.

CAs SHOULD also document whether they implement DNSSEC validation for DNS lookups done for validation purposes, as this affects the security of the "accounturi" and "validationmethods" parameters.

### 5.3. Mandatory Consistency in CA Recognition

A CA MUST ensure that its support for the "accounturi" and "validationmethods" parameters is fully consistent for a given domain name which a CA recognises as identifying itself in a CAA "issue" or "issuewild" property. If a CA has multiple issuance systems (for example, an ACME-based issuance system and a non-ACME based issuance system, or two different issuance systems resulting from a corporate merger), it MUST ensure that all issuance systems recognise the same parameters.

A CA which is unable to do this MAY still implement the parameters by splitting the CA into two domain names for the purposes of CAA processing. For example, a CA "example.com" with an ACME-based issuance system and a non-ACME-based issuance system could recognise only "acme.example.com" for the former and "example.com" for the

latter, and then implement support for the "accounturi" and "validationmethods" parameters for "acme.example.com" only.

A CA which is unable to ensure consistent processing of the "accounturi" or "validationmethods" parameters for a given CA domain name as specifiable in CAA "issue" or "issuewild" properties MUST NOT implement support for these parameters. Failure to do so will result in an implementation of these parameters which does not provide effective security.

### 5.4. URI Ambiguity

Suppose that CA A recognises "a.example.com" as identifying itself, CA B is a subsidiary of CA A which recognises both "a.example.com" and "b.example.com" as identifying itself.

Suppose that both CA A and CA B issue account URIs of the form

"account-id:1234"

If the CA domain name in a CAA record is specified as "a.example.com" then this could be construed as identifying account number 1234 at CA A or at CA B. These may be different accounts, creating ambiguity.

Thus, CAs MUST ensure that the URIs they recognise as pertaining to a specific account of that CA are unique within the scope of all domain names which they recognise as identifying that CA for the purpose of CAA record validation.

CAs MUST satisfy this requirement by using URIs which include an authority:

"https://a.example.com/account/1234"

### 5.5. Authorization Freshness

The CAA specification governs the act of issuance by a CA. In some cases, a CA may establish authorization for an account to request certificate issuance for a specific domain separately to the act of issuance itself. Such authorization may occur substantially prior to a certificate issuance request. The CAA policy expressed by a domain may have changed in the meantime, creating the risk that a CA will issue certificates in a manner inconsistent with the presently published CAA policy.

CAs SHOULD adopt practices to reduce the risk of such circumstances. Possible countermeasures include issuing authorizations with very

limited validity periods, such as an hour, or revalidating the CAA policy for a domain at certificate issuance time.

### 5.6. Use with and without DNSSEC

Where a domain chooses to secure its nameservers using DNSSEC, the authenticity of its DNS data can be assured, providing that a given CA makes all DNS resolutions via an appropriate, trusted DNSSEC-validating resolver. A domain can use this property to protect itself from the threat posed by a global adversary capable of performing man-in-the-middle attacks, which is not ordinarily mitigated by the "domain validation" model.

In order to facilitate this, a CA validation process must either rely solely on information obtained via DNSSEC, or meaningfully bind the other parts of the validation transaction using material obtained via DNSSEC.

The CAA parameters described in this specification can be used to ensure that only validation methods meeting these criteria are used. In particular, a domain secured via DNSSEC SHOULD either:

1. Use the "accounturi" parameter to ensure that only accounts which it controls are authorized to obtain certificates, or

2. Exclusively use validation methods which rely solely on information obtained via DNSSEC, and use the "validationmethods" parameter to ensure that only such methods are used.

Use of the "accounturi" or "validationmethods" parameters does not confer additional security against an attacker capable of performing a man-in-the-middle attack against all validation attempts made by a given CA which is authorized by CAA where:

1. A domain does not secure its nameservers using DNSSEC, or

2. That CA does not perform CAA validation using a trusted DNSSEC-validating resolver.

Moreover, use of the "accounturi" or "validationmethods" parameters does not mitigate against man-in-the-middle attacks against CAs which do not validate CAA records, or which do not do so using a trusted DNSSEC-validating resolver, regardless of whether those CAs are authorized by CAA or not; see Section 5.1.

In these cases, the "accounturi" and "validationmethods" parameters still provide an effective means of administrative control over issuance, except where control over DNS is subdelegated (see below).

## 5.7. Restrictions Supercedable by DNS Delegation

Because CAA records are located during validation by walking up the DNS hierarchy until one or more records are found, the use of the "accounturi" and "validationmethods" parameters, or any CAA policy, is not an effective way to restrict or control issuance for subdomains of a domain, where control over those subdomains is delegated to another party (such as via DNS delegation or by providing limited access to manage subdomain DNS records).

## 5.8. Misconfiguration Hazards

Because they express a restrictive security policy, misconfiguration of the "accounturi" or "validationmethods" parameters may result in legitimate issuance requests being refused.

## 6. IANA Considerations

None. As per the CAA specification, the parameter namespace for the CAA "issue" and "issuewild" properties has CA-defined semantics. This document merely specifies a RECOMMENDED semantic for parameters of the names "accounturi" and "validationmethods".

## 7. Normative References

[I-D.ietf-acme-acme]
Barnes, R., Hoffman-Andrews, J., McCarney, D., and J. Kasten, "Automatic Certificate Management Environment (ACME)", draft-ietf-acme-acme-18 (work in progress), December 2018.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <https://www.rfc-editor.org/info/rfc3986>.

[RFC6844] Hallam-Baker, P. and R. Stradling, "DNS Certification Authority Authorization (CAA) Resource Record", RFC 6844, DOI 10.17487/RFC6844, January 2013, <https://www.rfc-editor.org/info/rfc6844>.

## Appendix A.  Examples

The following shows an example DNS zone file fragment which nominates two account URIs as authorized to issue certificates for the domain "example.com".  Issuance is restricted to the CA "example.net".

```
example.com. IN CAA 0 issue "example.net; \
  accounturi=https://example.net/account/1234"
example.com. IN CAA 0 issue "example.net; \
  accounturi=https://example.net/account/2345"
```

The following shows a zone file fragment which restricts the ACME methods which can be used; only ACME methods "dns-01" and "xyz-01" can be used.

```
example.com. IN CAA 0 issue "example.net; \
  validationmethods=dns-01,xyz-01"
```

The following shows an equivalent way of expressing the same restriction:

```
example.com. IN CAA 0 issue "example.net; validationmethods=dns-01"
example.com. IN CAA 0 issue "example.net; validationmethods=xyz-01"
```

The following shows a zone file fragment in which one account can be used to issue with the "dns-01" method and one account can be used to issue with the "http-01" method.

```
example.com. IN CAA 0 issue "example.net; \
  accounturi=https://example.net/account/1234; \
  validationmethods=dns-01"
example.com. IN CAA 0 issue "example.net; \
  accounturi=https://example.net/account/2345; \
  validationmethods=http-01"
```

The following shows a zone file fragment in which only ACME method "dns-01" can be used, but non-ACME methods of issuance are also allowed.

```
example.com. IN CAA 0 issue "example.net; \
  validationmethods=dns-01,non-acme"
```

## Author's Address

Hugo Landau

Email: hlandau@devever.net