Automated Certificate Management Environment
Internet-Draft
Intended status: Standards Track
Expires: 4 September 2025

C. P. Liu
Huawei
M. Ounsworth
Entrust Limited
M. Richardson
Sandelman Software Works Inc

3 March 2025

Automated Certificate Management Environment (ACME) rats Identifier and Challenge Type
draft-liu-acme-rats-01

Abstract

This document describes an approach where an ACME Server can challenge an ACME Client to provide a Remote Attestation Evidence or Remote Attestation Result in any format supported by the Conceptual Message Wrapper. The ACME Server can optionally challenge the Client for specific claims that it wishes attestation for.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://liuchunchi.github.io/draft-liu-acme-rats/draft-liu-acme-rats.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-liu-acme-rats/.

Discussion of this document takes place on the Automated Certificate Management Environment Working Group mailing list (mailto:acme@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/acme/. Subscribe at https://www.ietf.org/mailman/listinfo/acme/.

Source for this draft and an issue tracker can be found at https://github.com/liuchunchi/draft-liu-acme-rats.

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/.

This Internet-Draft will expire on 4 September 2025.

Copyright Notice

Table of Contents

1. Introduction

ACME [RFC8555] is a standard protocol for issuing and renewing certificates automatically, widely used in the Internet scenario, help an ACME Client prove its ownership to an identifier like domain name or email address.

In order to prevent issuing certificates to malicious devices, a few works are ongoing in the LAMPS and RATS WG.

* [I-D.ietf-lamps-csr-attestation] define trustworthy claims about device's platform generating the certification signing requests (CSR) and the private key resides on this platform.

* [I-D.draft-moriarty-rats-posture-assessment] define a summary of a local assessment of posture for managed systems and across various layers.

This document builds on [I-D.draft-bweeks-acme-device-attest-01] which provides a mechanism for WebAuthn attestations over ACME. This document is broader in scope to support a broad range of attestation formats.

In this document, we propose an approach where ACME Server MAY challenge the ACME Client to produce an Attestation Evidence or Attestation Result in any format that is supported by the RATS Conceptual Message Wrapper [I-D.draft-ietf-rats-msg-wrap]. The ACME Server then checks if the ACME Clients presented a valid remote attestation evidence or remote attestation result, for instance, an EAT (entity attestation token). The ACME Server MAY perform any necessary checks against the provided remote attestation, as required by by the requested certificate profile; this conforms to the RATS concept of an Appraisal Policy.

This document defines a new ACME "rats" identifier and the challenge types "device-attest-02" and "device-attest-03" which are respectively use to challenge for a RATS background check and passport model type attestation. In this way, the CA / RA issues certificates only to devices that can provide an appropriate attestation result, indicating such device has passed the required security checks. By repeating this process and issuing only short-lived certificates to qualified devices, we also fulfill the continuous monitoring/validation requirement of Zero-Trust principle.

Some example use cases include an enterprise scenario where Network Operations Center (NOC) issue certificates to devices that can prove via remote attestation that they are running an up-to-date operating system as well as the enterprise-required endpoint security software. Another example is issuing S/MIME certificates to BYOD devices only if they can prove via attestation that they are registered to a corporate MDM and the user they are registered to matches the user for which a certificate has been requested.

For ease of denotion, we omit the "ACME" adjective from now on, where Server means ACME Server and Client means ACME Client.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Extensions -- rats identifier

An rats identifier type represents a unique identifier to an attestation result. It extends a "rats" identifier type and a string value.

type (required, string): The string "rats".

value (required, string): The identifier itself.

The following steps are the ones that will be affected:

1. newOrder Request Object - identifiers: During the certificate order creation step, the Client sends a /newOrder JWS request (Section 7.4 of [RFC8555]) whose payload contains an array of identifiers. The Client adds an rats identifier to the array.

An example extended newOrder JWS request:

```
{
  "protected": base64url({
    "alg": "ES256",
  }),
  "payload": base64url({
    "identifiers": [
      { "type": "rats", "value": "0123456789abcdef" },
    ],
  }),
  "signature": "H6ZXtGjTZyUnPeKn...wEA4TklBdh3e454g"
}
```

2. Order Object - identifiers: After a newOrder request is sent to the Server, the Account Object creates an Order Object (Section 7.1.3 of [RFC8555]) with "rats" identifiers and values from Step 1.

An example extended Order Object:

```
    {
      "status": "pending",

      "identifiers": [
        { "type": "rats", "value": "0123456789abcdef" },
      ],

      "authorizations": [
        "https://example.com/acme/authz/PAniVnsZcis",
      ],

      "finalize": "https://example.com/acme/order/T..fgo/finalize",
    }
```

3. Authorization Object - identifier: The Server creates an Authorization Object that has rats identifier (Section 7.1.4 of [RFC8555])

4. Challenge Object - identifier: The Server creates a Challenge Object that has rats challenge type.

An example extended Authorization Object (that contains a Challenge Object):

```
{
  "status": "pending",

  "identifier": {
    "type": "rats",
    "value": "0123456789abcdef"
  },

  "challenges": [
    {
      "type": "rats",
      "url": "https://example.com/acme/chall/prV_B7yEyA4",
      "status": "pending",
      "token": "DGyRejmCefe7v4NfDGDKfA",
    },
    {
      "type": "http-01",
      "url": "https://example.com/acme/chall/prV_B7yEyA4",
      "status": "pending",
      "token": "DGyRejmCefe7v4NfDGDKfA",
    }
  ],
}
```

4. Extensions -- rats challenge types

A rats challenge type help the Client prove ownership to its attestation result identifier. This section describes the challenge/response extensions and procedures to use them.

4.1. device-attest-02 Challenge

device-attest-02 Challenge simply works with Passport Model of RATS. The corresponding Challenge Object is:

type (required, string): The string "device-attest-02".

url (required, string): The URL that the Client post its response to.

token (required, string): Same as Section 8.3 of RFC8555.

nonce (optional, string): If attestation freshness is required, then the Server MAY present a nonce which then MUST be echoed in the provided attestation. In some situations, the nonce will come from a separate RATS Verifier, and therefore needs to be a distinct value from the ACME token.

attestClaimsHint (optional, list of string) If the Server requires attestation of specific claims or properties in order to issue the requested certificate profile, then it MAY list one or more types of claims from the newly-defined ACME Attest Claims Hints registry defined in "sec-claimshints"(cite-fail).

The response sent to the url is:

keyAuthorization = token || '.' || cmw

where cmw MAY be either a CMW in JWT format, or a Base64 CMW in CWT format as per [I-D.draft-ietf-rats-msg-wrap].

4.2. device-attest-03 Challenge

device-attest-03 Challenge works the same way as device-attest-02, but expects the Client to return RATS evidence in accordance with the Background Check Model of RATS.

5. ACME Attest Claims Hint Registry

In order to facilitate the Server requesting attestation of specific types claims or properties, we define a new registry of ACME Claims Hints. In order to preserve flexibility, the Claim Hints are intended to be generic in nature, allowing for the client to reply with any type of attestation evidence or attestation result that contains the requested information. As such, these values are not intended to map one-to-one with any specific remote attestation evidence or attestation result format, but instead they are to serve as a hint to the ACME Client about what type of attestation it needs to collect from the device. Ultimately, the CA's certificate policies will be the authority on what evidence or attestation results it will accept.

See Section 8.1 for the initial contents of this new registry.

6. Example use case -- enterprise access

In an enterprise network scenario, it is hard to coordinate Security Operations Center (SOC) and Network Operations Center (NOC) to work together due to various of reasons:

1. Integration/compatibility difficulty: Integrating SOC and NOC requires plenty of customized, case-by-case developing work. Especially considering differrnt system vendors, system versions, different data models and formats due to different client needs... Let alone possible updates.

2. Conflict of duties: NOC people do not want SOC people to interfere with their daily work, and so do SOC people. Also, NOC people may have limited security knowledge, and SOC people vice versa. Where to draw the line and what is the best tool to help them collaborate is a question.

This work proposes a way to help SOC and NOC work together, with separated duties (to avoid conflict) and ease of working together (proper abstraction).

An Endpoint Detection and Response (EDR) software and Security Operations Center (SOC) is responsible for checking the security status of an accessing end device. If the device passed latest security checks, EDR/SOC should issue fresh attestation results (consider as a security clearance). Otherwise, EDR/SOC should refuse to issue (new) attestation results. A Network Operations Center (NOC) could use ACME to issue short-lived certificates to only devices with fresh attestation results. In this way, the NOC can follow a Zero-Trust philosophy and issue network access to only

devices that are continuously monitored and have no known security risks up-to-date. SOC can also have flexible security policies and decide what to check.

7. Security Considerations

TODO Security

8. IANA Considerations

IANA is requested to open a new registry, XXXXXXXX

Type: designated expert

The registry has the following columns:

* Claim Hint: the string value to be placed within an ACME device-attest-02 or device-attest-03 challenge.

* Descryption: a descryption of the general type of attestation evidence or attestation result that the client is expected to produce.

The initial registry contents is shown in the table below.

Claim Hint	Description
FIPS_mode	Attestation that the device is currently booted in FIPS mode.
OS_patch_level	Attestation to the version or patch level of the device’s operating system.
sw_manifest	A manifest list of all software currently running on the device.

Table 1

8.1. ACME Attest Claims Hint Registry

9. References

9.1. Normative References

[RFC8555] Barnes, R., Hoffman-Andrews, J., McCarney, D., and J. Kasten, "Automatic Certificate Management Environment (ACME)", RFC 8555, DOI 10.17487/RFC8555, March 2019, <https://www.rfc-editor.org/rfc/rfc8555>.

[I-D.draft-ietf-rats-msg-wrap] Birkholz, H., Smith, N., Fossati, T., Tschofenig, H., and D. Glaze, "RATS Conceptual Messages Wrapper (CMW)", Work in Progress, Internet-Draft, draft-ietf-rats-msg-wrap-12, 28 February 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-rats-msg-wrap-12>.

9.2. Informative References

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[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/rfc/rfc2119>.

[I-D.ietf-lamps-csr-attestation] Ounsworth, M., Tschofenig, H., Birkholz, H., Wiseman, M., and N. Smith, "Use of Remote Attestation with Certification Signing Requests", Work in Progress, Internet-Draft, draft-ietf-lamps-csr-attestation-17, 2 March 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-lamps-csr-attestation-17>.

[I-D.draft-moriarty-rats-posture-assessment] Moriarty, K., Wiseman, M., and A. Stein, "Remote Posture Assessment for Systems, Containers, and Applications", Work in Progress, Internet-Draft, draft-moriarty-rats-posture-assessment-01, 2 July 2024, <https://datatracker.ietf.org/doc/html/draft-moriarty-rats-posture-assessment-01>.

[I-D.draft-bweeks-acme-device-attest-01] Weeks, B., "Automated Certificate Management Environment (ACME) Device Attestation Extension", Work in Progress, Internet-Draft, draft-bweeks-acme-device-attest-01, 7 August 2022, <https://datatracker.ietf.org/doc/html/draft-bweeks-acme-device-attest-01>.

Acknowledgments

TODO acknowledge.

Authors' Addresses

Chunchi Peter Liu
Huawei
Email: liuchunchi@huawei.com

Mike Ounsworth
Entrust Limited
Email: mike.ounsworth@entrust.com

Michael Richardson
Sandelman Software Works Inc
Email: mcr+ietf@sandelman.ca