Automated Certificate Management Environment
Internet-Draft
Intended status: Experimental
Expires: 16 April 2022

B. Sipos
RKF Engineering
13 October 2021

Automated Certificate Management Environment (ACME) Delay-Tolerant Networking (DTN) Node ID Validation Extension
draft-ietf-acme-dtnnodeid-06

Abstract

This document specifies an extension to the Automated Certificate Management Environment (ACME) protocol which allows an ACME server to validate the Delay-Tolerant Networking (DTN) Node ID for an ACME client. The DTN Node ID is encoded as a certificate Subject Alternative Name (SAN) of type otherName with a name form of BundleEID and as an ACME Identifier type "bundleEID".

Table of Contents

1. Introduction

Although the original purpose of the Automatic Certificate Management Environment (ACME) [RFC8555] was to allow Public Key Infrastructure Using X.509 (PKIX) certificate authorities to validate network domain names of clients, the same mechanism can be used to validate any of the subject claims supported by the PKIX profile [RFC5280].

In the case of this specification, the claim being validated is a Subject Alternative Name (SAN) of type otherName with a name form of BundleEID, which used to represent an Endpoint ID (EID) for a Delay-Tolerant Networking (DTN) bundle. Currently the URI schemes "dtn" and "ipn" as defined in [I-D.ietf-dtn-bpbis] are valid for an Endpoint ID. A DTN Node ID is an Endpoint ID with scheme-specific restrictions to identify it as such; currently the "dtn" scheme uses an empty demux part and the "ipn" scheme uses service number zero.

Because the BundleEID claim is new to ACME, a new ACME Identifier type "bundleEID" is needed to manage this claim within ACME messaging. A "bundleEID" claim can be part of a pre-authorization or as one of the authorizations of an order containing any number of claims.

Once an ACME server validates a Node ID, either as a pre-authorization of the "bundleEID" or as one of the authorizations of an order containing a "bundleEID", the client can finalize the order using an associated certificate signing request (CSR). Because a single order can contain multiple identifiers of multiple types, there can be operational issues for a client attempting to, and possibly failing to, validate those multiple identifiers as described in Section 5.1. Once a certificate is issued for a Node ID, how the ACME client configures the Bundle Protocol (BP) agent with the new certificate is an implementation matter.

The scope and behavior of this validation mechanism is similar to that of secured email validation of [RFC8823].

1.1. Scope

This document describes the ACME messages, BPv7 payloads, and BPSec requirements needed to validate Node ID ownership. This document does not address:

* Mechanisms for communication between ACME client or ACME server and their associated BP agent(s). This document only describes exchanges between ACME client--server pairs and between their BP agents.

* Specific BP extension blocks or BPSec security contexts necessary to fulfill the security requirements of this protocol. The exact security context needed, and their parameters, are network-specific.

* Policies or mechanisms for defining or configuring bundle integrity gateways, or trusting integrity gateways on an individual entity or across a network.

* Mechanisms for locating or identifying other bundle entities (peers) within a network or across an internet. The mapping of Node ID to potential convergence layer (CL) protocol and network address is left to implementation and configuration of the BP Agent and its various potential routing strategies.

* Logic for routing bundles along a path toward a bundle's endpoint. This protocol is involved only in creating bundles at a source and handling them at a destination.

* Logic for performing rate control and congestion control of bundle transfers. The ACME server is responsible for rate control of validation requests.

* Policies or mechanisms for provisioning, deploying, or accessing certificates and private keys; deploying or accessing certificate revocation lists (CRLs); or configuring security parameters on an individual entity or across a network.

* Policies or mechanisms for an ACME server to handle mixed-use certificate requests. This specification is focused only on single-use DTN-specific PKIX profiles.

1.2. Authorization Strategy

The basic unit of data exchange in a DTN is a Bundle [I-D.ietf-dtn-bpbis], which consists of a data payload with accompanying metadata. An Endpoint ID is used as the destination of a Bundle and can indicate both a unicast or a multicast destination. A Node ID is used to identify the source of a Bundle and is used for routing through intermediate nodes, including the final node(s) used to deliver a Bundle to its destination endpoint. A Node ID can also be used as an endpoint for administrative bundles. More detailed descriptions of the rationale and capabilities of these networks can be found in "Delay-Tolerant Network Architecture" [RFC4838].

When an ACME client requests a pre-authorization or an order with a "bundleEID" identifier type having a value consistent with a Node ID (see Section 4.2.5 of [I-D.ietf-dtn-bpbis]), the ACME server offers a "dtn-nodeid-01" challenge type to validate that Node ID. If the ACME client attempts the authorization challenge to validate a Node ID, the ACME server sends an ACME Node ID Validation Challenge Bundle with a destination of the Node ID being validated. The BP agent on that node receives the Challenge Bundle, generates an ACME key authorization digest, and sends an ACME Node ID Validation Response Bundle in reply. An Integrity Gateway on the client side of the DTN can be used to attest to the source of the Response Bundle. Finally, the ACME server receives the Response Bundle and checks that the digest was generated for the associated ACME challenge and from the client account key associated with the original request. This workflow is shown in the diagram of Figure 1 and defined in Section 3.

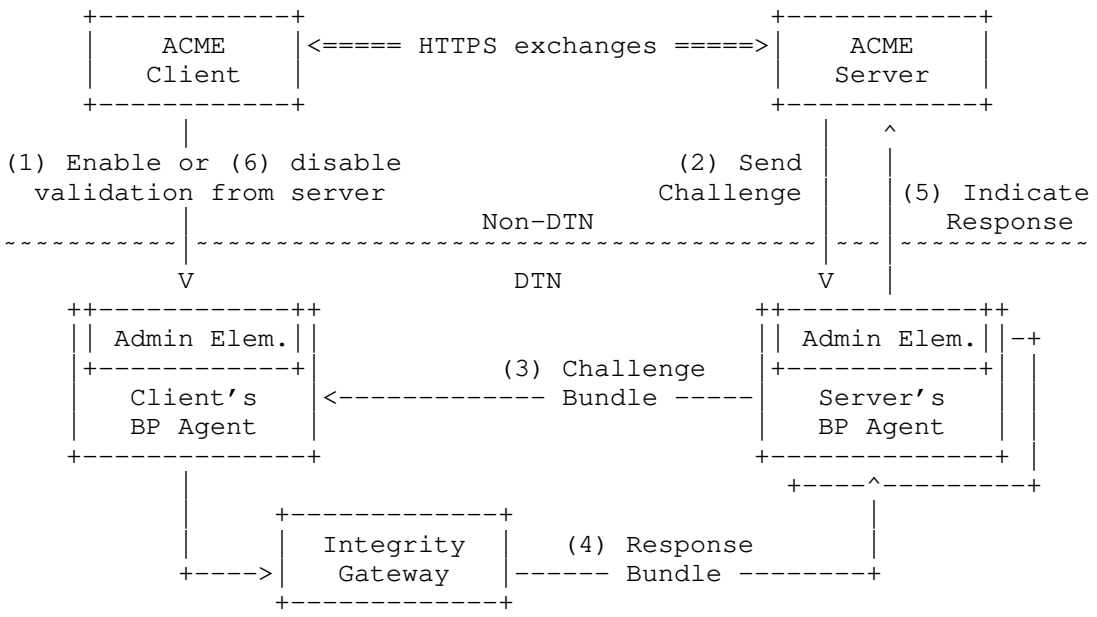

Figure 1: The relationships and flows between Node ID Validation entities

Because the DTN Node ID is used both for routing bundles between BP agents and for multiplexing administrative services within a BP agent, there is no possibility to separate the ACME validation of a Node ID from normal bundle handling for that same Node ID. This leaves administrative record types as a way to leave the Node ID unchanged while disambiguating from other service data bundles.

There is nothing in this protocol which requires network-topological co-location of either the ACME client or ACME server with their associated BP agent. While ACME requires a low-enough latency network to perform HTTPS exchanges between ACME client and server, the client's BP agent (the one being validated) could be on the far side of a long-delay or multi-hop DTN network. The means by which the ACME client or server communicates with its associated BP agent is an implementation matter.

1.3. Use of CDDL

This document defines CBOR structure using the Concise Data Definition Language (CDDL) of [RFC8610]. The entire CDDL structure can be extracted from the XML version of this document using the XPath expression:

'//sourcecode[@type="cddl"]'

The following initial fragment defines the top-level symbols of this document's CDDL, which includes the example CBOR content.

```
start = acme-record / bundle / tstr
```

1.4. 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.

In this document, several terms are shortened for the sake of terseness. These terms are:

Challenge Request: This is a shortened form of the full "DTN Node ID Challenge Request Object". It is a JSON object created by the ACME server for challenge type "dtn-nodeid-01".

Challenge Response: This is a shortened form of the full "DTN Node ID Challenge Response Object". It is a JSON object created by the ACME client to authorize a challenge type "dtn-nodeid-01".

Challenge Bundle: This is a shortened form of the full "ACME Node ID Validation Challenge Bundle". It is a Bundle created by the BP agent managed by the ACME server to challenge a Node ID claim.

Response Bundle: This is a shortened form of the full "ACME Node ID Validation Response Bundle". It is a Bundle created by the BP agent managed by the ACME client to validate a Node ID claim.

2. Bundle Endpoint ID ACME Identifier

This specification is the first to make use of an Bundle Endpoint ID to identify a claim for a certificate request in ACME. In this document, the only purpose for which an Bundle Endpoint ID ACME identifier is validated is as a DTN Node ID (see Section 3), but other specifications can define challenge types for other Endpoint ID uses.

Identifiers of type "bundleEID" in certificate requests MUST appear in an extensionRequest attribute [RFC2985] containing a subjectAltName extension of type otherName with a name form of BundleEID, identified by id-on-bundleEID of [IANA-SMI], consistent with the requirements of Section 4.4.2.1 of [I-D.ietf-dtn-tcpclv4].

Because the BundleEID is encoded as an IA5String it SHALL be treated as being in the percent-encoded form of Section 2.1 of [RFC3986]. Any "bundleEID" identifier which fails to properly percent-decode SHALL be rejected with an ACME error type of "malformed".

The ACME server SHALL decode and normalize (based on scheme-specific syntax) all received identifiers of type "bundleEID". Any "bundleEID" identifier request which uses a scheme not handled by the ACME server or for which the EID does not match the scheme-specific syntax SHALL be rejected with an ACME error type of "rejectedIdentifier".

When an ACME server needs to request proof that a client controls a BundleEID, it SHALL create an authorization with the identifier type "bundleEID". The ACME server SHALL NOT attempt to dereference the EID value on its own. It is the responsibility of a validation method to ensure the EID ownership via scheme-specific means authorized by the ACME client.

An identifier for the Node ID of "dtn://example/" would be formatted as:

```
{
  "type": "bundleEID",
  "value": "dtn://example/"
}
```

3. DTN Node ID Validation

The DTN Node ID validation method proves control over a Node ID by requiring the ACME client to configure a BP agent to respond to specific Challenge Bundles sent from the ACME server. The ACME server validates control of the Node ID by verifying that received Response Bundles correspond with the BP Node and client account key being validated.

Similar to the ACME use case for validating email address ownership [RFC8823], this challenge splits the token into several parts, each being transported by a different channel, and the Key Authorization result requires combining all parts of the token. The token parts are:

token-chal This token is unique to, and identifies, each ACME authorization. It is contained in the Challenge Object of Section 3.1 as well as the Challenge Bundle of Section 3.3 and Response Bundle of Section 3.4. Each authorization can consist of multiple Challenge Bundles (e.g. taking different routes), but they all share the same token-chal value. This ensures that the

Key Authorization is bound to the specific ACME challenge (and parent ACME authorization) and also allows the ACME client's BP agent to filter-in only valid Challenge Bundles. This token is also accessible to DTN on-path eavesdroppers.

token-bundle This token is unique to each Challenge Bundle sent by the ACME server. It is contained in the Challenge Bundle of Section 3.3 and Response Bundle of Section 3.4. This ensures that the Key Authorization is bound to the ability to receive the Challenge Bundle and not just have access to the ACME Challenge Object. This token is also accessible to DTN on-path eavesdroppers.

For each ACME server, the pair of token-chal and token-bundle values is the unique correlator between Challenge and Response bundles. Because multiple Challenge Bundles can be sent to validate the same Node ID, the token-bundle in the response is needed to correlate with the expected Key Authorization digest.

The DTN Node ID Challenge SHALL only be allowed for an EID usable as a DTN Node ID, which [I-D.ietf-dtn-bpbis]. When an ACME server supports Node ID validation, the ACME server SHALL define a challenge object in accordance with Section 3.1. Once this challenge object is defined, the ACME client may begin the validation.

To initiate a Node ID validation, the ACME client performs the following steps:

1. The ACME client POSTs a newOrder or newAuthz request including the identifier of type "bundleEID" for the desired Node ID. From either of these entry points an authorization for the "bundleEID" type is indicated by the ACME server. See Section 7.4 of [RFC8555] for more details.

2. The ACME client obtains the challenge source Node ID and token-chal from the challenge object in accordance with Section 3.1.

3. The ACME client indicates to the administrative element of its BP agent the source Node ID and challenge token-chal which is authorized for use and the associated client account key thumbprint. The ACME client SHALL wait, if necessary, until the agent is configured before proceeding to the next step.

4. The ACME client POSTs a challenge response to the challenge URL on the ACME server accordance with Section 7.5.1 of [RFC8555]. The payload object is constructed in accordance with Section 3.2.

5. The administrative element waits for a Challenge Bundle to be received with the authorized ACME parameters, including its token-bundle payload, in accordance with Section 3.3.

6. The administrative element concatenates token-bundle with token-chal (each as base64url-encoded text strings) and computes the Key Authorization in accordance with Section 8.1 of [RFC8555] using the full token and client account key thumbprint.

7. The administrative element computes the SHA-256 digest of the Key Authorization result and generates a Response Bundle to send back to the ACME server in accordance with Section 3.4.

8. The ACME client waits for the authorization to be finalized on the ACME server in accordance with Section 7.5.1 of [RFC8555].

9. Once the challenge is completed (successfully or not), the ACME client indicates to the BP agent that the validation source and token-chal is no longer usable. If the authorization fails, the ACME client MAY retry the challenge from Step 3.

The ACME server verifies the client's control over a Node ID by performing the following steps:

1. The ACME server receives a newOrder or newAuthz request including the identifier of type "bundleEID", where the URI value is a Node ID.

2. The ACME server generates an authorization for the Node ID with challenge type "dtn-nodeid-01" in accordance with Section 3.1.

3. The ACME server receives a POST to the challenge URL indicated from the authorization object. The payload is handled in accordance with Section 3.2.

4. The ACME server sends, via the administrative element of its BP agent, one or more Challenge Bundles in accordance with Section 3.3. Each challenge bundle SHALL contain a distinct token-bundle to be able to correlate with a response bundle. Computing an expected Key Authorization digest is not necessary until a response is received.

5. The ACME server waits for Response Bundle(s) for a limited interval of time (based on the challenge response object of Section 3.2). Responses are encoded in accordance with Section 3.4.

6. Once received and decoded, the ACME server checks the contents of each Response Bundle in accordance with Section 3.4.1. After all Challenge Bundles have either been responded to or timed-out, the ACME server policy (see Section 3.5) determines whether the validation is successful. If validation is not successful, a client may retry the challenge which starts in Step 3.

When responding to a Challenge Bundle, a BP agent SHALL send a single Response Bundle in accordance with Section 3.4. A BP agent SHALL respond to ACME challenges only within the interval of time, only for the Node ID, and only for the token-chal indicated by the ACME client. A BP agent SHALL respond to multiple Challenge Bundles with the same ACME parameters but different bundle identities (source Node ID and timestamp); these correspond with the ACME server validating via multiple routing paths.

3.1. DTN Node ID Challenge Request Object

The DTN Node ID Challenge request object is defined by the ACME server when it supports validating Node IDs.

The DTN Node ID Challenge request object has the following content:

type (required, string): The string "dtn-nodeid-01".

source (required, array of string): An unordered list of possible source Node ID of bundles originating at the BP agent(s) of the ACME server. See Section 3.5 for a discussion of multi-perspective validation using multiple sources. The array SHALL be non-empty. The array MAY contain Node IDs which are not actually used as a challenge bundle source.

token-chal (required, string): A random value that uniquely identifies the challenge. This value MUST have at least 128 bits of entropy. It MUST contain any characters outside the base64url alphabet as described in Section 5 of [RFC4648]. Trailing '=' padding characters MUST be stripped. See [RFC4086] for additional information on randomness requirements.

```
{
  "type": "dtn-nodeid-01",
  "url": "https://example.com/acme/chall/prV_B7yEyA4",
  "source": ["dtn://acme-server/"],
  "token-chal": "tPUZNY4ONIk6LxErRFEjVw"
}
```

The token-chal value included in this object is fixed for the entire challenge, and may correspond with multiple separate token-bundle values when multiple Challenge Bundles are sent for a single validation.

3.2. DTN Node ID Challenge Response Object

The DTN Node ID Challenge response object is defined by the ACME client when it authorizes validation of a Node ID. Because a DTN has the potential for significantly longer delays than a non-DTN network, the ACME client is able to inform the ACME server if a particular validation round-trip is expected to take longer than normal network delays (on the order of seconds).

The DTN Node ID Challenge response object has the following content:

rtt (optional, number): An expected round-trip time (RTT), in seconds, between sending a Challenge Bundle and receiving a Response Bundle. This value is a hint to the ACME server for how long to wait for responses but is not authoritative. The minimum RTT value SHALL be zero. There is no special significance to zero-value RTT, it simply indicates the response is expected in less than the least significant unit used by the ACME client.

```
{
  "rtt": 300.0
}
```

A challenge response is not sent until the BP agent has been configured to properly respond to the challenge, so the RTT value is meant to indicate any node-specific path delays expected to encountered from the ACME server. Because there is no requirement on the path (or paths) which bundles may traverse between the ACME server and the BP agent, and the ACME server can attempt some path diversity, the RTT value SHOULD be pessimistic.

A default bundle response interval, used when the object does not contain an RTT, SHOULD be a configurable parameter of the ACME server. If the ACME client indicated an RTT value in the object, the response interval SHOULD be twice the RTT (with limiting logic applied as described below). The lower limit on response interval is network-specific, but SHOULD NOT be shorter than one second. The upper limit on response interval is network-specific, but SHOULD NOT be longer than one minute (60 seconds) for a terrestrial-only DTN.

3.3. ACME Node ID Validation Challenge Bundles

Each ACME Node ID Validation Challenge Bundle SHALL be structured and encoded in accordance with [I-D.ietf-dtn-bpbis].

Each Challenge Bundle has parameters as listed here:

Bundle Processing Control Flags: The primary block flags SHALL indicate that the payload is an administrative record. The primary block flags SHALL indicate that user application acknowledgement is requested; this flag distinguishes the Challenge Bundle from the Response Bundle. The primary block flags MAY indicate that status reports are requested; such status can be helpful to troubleshoot routing issues.

Destination EID: The Destination EID SHALL be the ACME-server-normalized Node ID being validated.

Source Node ID: The Source Node ID SHALL indicate the Node ID of the BP agent of the ACME server performing the challenge. The challenge bundle source SHALL be present in the "source" array of the challenge object (see Section 3.1)

Creation Timestamp and Lifetime: The Creation Timestamp SHALL be set to the time at which the challenge was generated. The Lifetime SHALL indicate the response interval (of Section 3.2) for which ACME server will accept responses to this challenge.

Administrative Record Type Code: Set to the ACME Node ID Validation type code defined in Section 8.3.

Administrative Record Content: The Challenge Bundle administrative record content SHALL consist of a CBOR map containing two pairs:

* One pair SHALL consist of key 1 with value of ACME challenge token-chal, copied from the challenge object, represented as a CBOR byte string.

* One pair SHALL consist of key 2 with value of ACME challenge token-bundle, represented as a CBOR byte string. The token-bundle is a random value that uniquely identifies the challenge bundle. This value MUST have at least 128 bits of entropy. See [RFC4086] for additional information on randomness requirements.

This structure is part of the extension CDDL in Appendix A. An example full Challenge Bundle is included in Appendix B.1

If the BP agent generating a Challenge Bundle does not have a well-synchronized clock or the agent responding to the challenge is expected to not have a well-synchronized clock, the bundle SHALL include a Bundle Age extension block.

Challenge Bundles SHALL include a Block Integrity Block (BIB) in accordance with Section 4 or with a Security Source identical to the bundle Source Node. Challenge Bundles SHALL NOT be directly encrypted by Block Confidentiality Block (BCB) or any other method (see Section 7.1).

3.3.1. Challenge Bundle Checks

A proper Challenge Bundle meets all of the following criteria:

* The Challenge Bundle was received within the time interval allowed for the challenge. The allowed interval starts at the Creation Timestamp and extends for the Lifetime of the Challenge Bundle.

* The Challenge Bundle Source Node ID is identical to the Node ID indicated in the ACME challenge object. The comparison of Node IDs SHALL use the comparison logic of the NODE-ID from Section 4.4.1 of [I-D.ietf-dtn-tcpclv4].

* The Challenge Bundle contains a BIB which covers at least the primary block and payload. That BIB has a security source which is trusted and passes security-context-specific validation (i.e. MAC or signature verification).

* The challenge payload contains the token-chal as indicated in the ACME challenge object. The challenge payload contains a token-bundle meeting the definition in Section 3.3.

Any of the failures above SHALL cause the challenge bundle to be deleted and otherwise ignored by the BP agent. The BP agent MAY send status reports about the deletion if allowed by security policy.

3.4. ACME Node ID Validation Response Bundles

Each ACME Node ID Validation Response Bundle SHALL be structured and encoded in accordance with [I-D.ietf-dtn-bpbis].

Each Response Bundle has parameters as listed here:

Bundle Processing Control Flags: The primary block flags SHALL indicate that the payload is an administrative record. The primary block flags SHALL NOT indicate that user application acknowledgement is requested; this flag distinguishes the Response

Bundle from the Challenge Bundle. The primary block flags MAY indicate that status reports are requested; such status can be helpful to troubleshoot routing issues.

Destination EID: The Destination EID SHALL be identical to the Source Node ID of the Challenge Bundle to which this response corresponds.

Source Node ID: The Source Node ID SHALL be identical to the Destination EID of the Challenge Bundle to which this response corresponds.

Creation Timestamp and Lifetime: The Creation Timestamp SHALL be set to the time at which the response was generated. The response Lifetime SHALL indicate the response interval remaining if the Challenge Bundle indicated a limited Lifetime.

Administrative Record Type Code: Set to the ACME Node ID Validation type code defined in Section 8.3.

Administrative Record Content: The Response Bundle administrative record content SHALL consist of a CBOR map containing three pairs:

* One pair SHALL consist of key 1 with value of ACME challenge token-chal, copied from the Request Bundle, represented as a CBOR byte string.

* One pair SHALL consist of key 2 with value of ACME challenge token-bundle, copied from the Request Bundle, represented as a CBOR byte string.

* One pair SHALL consist of key 3 with value of the SHA-256 digest [FIPS180-4] of the ACME Key Authorization in accordance with Section 8.1 of [RFC8555], represented as a CBOR byte string.

This structure is part of the extension CDDL in Appendix A. An example full Response Bundle is included in Appendix B.2

If the BP agent responding to a Challenge Bundle does not have a well-synchronized clock, it SHALL use any information about last-hop delays and (if present) Bundle Age extension data to infer the age of the Challenge Bundle and lifetime of the Response Bundle.

Response Bundles SHALL include a BIB in accordance with Section 4. Response Bundles SHALL NOT be directly encrypted by BCB or any other method (see Section 7.1 for explanation).

3.4.1. Response Bundle Checks

A proper Response Bundle meets all of the following criteria:

* The Response Bundle was received within the time interval allowed for the challenge. The allowed interval starts at the Creation Timestamp and extends for the Lifetime of the associated Challenge Bundle. The interval of the Challenge Bundle is used here because the interval of the Response Bundle could be made to disagree with the Challenge Bundle.

* The Response Bundle Source Node ID is identical to the Node ID being validated. The comparison of Node IDs SHALL use the comparison logic of the NODE-ID from Section 4.4.1 of [I-D.ietf-dtn-tcpclv4].

* The Response Bundle contains a BIB which covers at least the primary block and payload. That BIB has a security source which is trusted and passes security-context-specific validation.

* The response payload contains the same token-chal and token-bundle as sent in the Challenge Bundle (this is also how the two bundles are correlated). The response payload contains the expected Key Authorization digest computed by the ACME server.

Any of the failures above SHALL cause that single-perspective validation to fail. Any of the failures above SHOULD be distinguished as subproblems to the ACME client. The lack of a response within the expected response interval, as defined in Section 3.2, SHALL also be treated as a validation failure.

3.5. Multi-Perspective Validation

To avoid possible on-path attacks in certain networks, an ACME server can perform a single validation using multiple challenge bundle sources or via multiple routing paths. This technique is called multi-perspective validation as recommended in Section 10.2 of [RFC8555] and an implementation used by Let's Encrypt is described in [LE-multi-perspective].

When required by policy, an ACME server SHALL send multiple challenge bundles from different sources in the DTN network. When multiple Challenge Bundles are sent for a single validation, it is a matter of ACME server policy to determine whether or not the validation as a whole is successful. The result of each single-source validation is defined as success or failure in Section 3.4.1.

A RECOMMENDED validation policy is to succeed if the challenge from a primary bundle source is successful and if there are no more than one failure from a secondary source. The determination of which perspectives are considered primary or secondary is an implementation matter.

Regardless of whether a validation is single- or multi-perspective, a validation failure SHALL be indicated by an ACME error type of "incorrectResponse". Each specific perspective failure SHOULD be indicated to the ACME client as a validation subproblem.

4. Bundle Integrity Gateway

This section defines a BIB use which closely resembles the function of DKIM email signing [RFC6376]. In this mechanism a routing node in a DTN sub-network attests to the origination of a bundle by adding a BIB before forwarding it. The bundle receiver then need not trust the source of the bundle, but only trust this security source node. The receiver needs policy configuration to know which security sources are permitted to attest for which bundle sources.

An integrity gateway SHALL validate the Source Node ID of a bundle, using local-network-specific means, before adding a BIB to the bundle. The exact means by which an integrity gateway validates a bundle's source is network-specific, but could use physical-layer, network-layer or BP-convergence-layer authentication. The bundle source could also add its own BIB with a local-network-specific security context or local-network-specific key material (i.e. a group key shared within the local network).

When an integrity gateway adds a BIB it SHALL be in accordance with [I-D.ietf-dtn-bpsec]. The BIB targets SHALL cover both the payload block and the primary block (either directly as a target or as additional authenticated data for the payload block MAC/signature). The Security Source of this BIB SHALL be either the bundle source Node ID itself or a routing node trusted by the destination node (see Section 7.2).

5. Certificate Request Profile

The ultimate purpose of this ACME validation is to allow a CA to issue certificates following the profiles of Section 4.4.2 of [I-D.ietf-dtn-tcpclv4], [I-D.sipos-dtn-udpcl], and [I-D.bsipos-dtn-bpsec-cose]. These purposes are referred to here as bundle security certificates.

One defining aspect of bundle security certificates is the Extended Key Usage key purpose id-kp-bundleSecurity of [IANA-SMI]. When requesting a certificate which includes a Node ID SAN, the CSR SHOULD include an Extended Key Usage of id-kp-bundleSecurity. When a bundle security certificate is issued based on a validated Node ID SAN, the certificate SHALL include an Extended Key Usage of id-kp-bundleSecurity.

5.1. Multiple Identity Claims

A single bundle security CSR MAY contain a mixed set of SAN claims, including combinations of "ip", "dns", and "bundleEID" claims. There is no restriction on how a certificate combines these claims, but each claim MUST be validated by an ACME server to issue such a certificate as part of an associated ACME order. This is no different than the existing behavior of [RFC8555] but is mentioned here to make sure that CA policy handles such situations; especially related to validation failure of an identifier in the presence of multiple identifiers. The specific use case of [I-D.ietf-dtn-tcpclv4] allows, and for some network policies requires, that a certificate authenticate both the DNS name of an entity as well as the Node ID of the entity.

5.2. Generating Encryption-only or Signing-only Bundle Security Certificates

ACME extensions specified in this document can be used to request encryption-only or signing-only bundle security certificates.

In order to request signing only bundle security certificate, the CSR MUST include the key usage extension with digitalSignature and/or nonRepudiation bits set and no other bits set.

In order to request encryption only bundle security certificate, the CSR MUST include the key usage extension with keyEncipherment or keyAgreement bits set and no other bits set.

Presence of both of the above sets of key usage bits in the CSR, as well as absence of key usage extension in the CSR, signals to ACME server to issue a bundle security certificate suitable for both signing and encryption.

6. Implementation Status

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

[NOTE to the RFC Editor: please remove this section before publication, as well as the reference to [RFC7942] and [github-dtn-demo-agent] and [github-dtn-wireshark].]

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in [RFC7942]. The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations can exist.

An example implementation of the this draft of ACME Node ID Validation has been created as a GitHub project [github-dtn-demo-agent] and is intended to use as a proof-of-concept and as a possible source of interoperability testing.

A Wireshark dissector for of the this draft of ACME Node ID Validation has been created as a GitHub project [github-dtn-wireshark] and is intended to be used to inspect and troubleshoot implementations.

7. Security Considerations

This section separates security considerations into threat categories based on guidance of BCP 72 [RFC3552].

7.1. Threat: Passive Leak of Validation Data

Because this challenge mechanism is used to bootstrap security between DTN Nodes, the challenge and its response are likely to be transferred in plaintext. The only ACME data present on-the-wire is a random token and a cryptographic digest, so there is no sensitive data to be leaked within the Node ID Validation bundle exchange. Because each challenge uses a separate token, there is no value in an on-path attacker seeing the tokens from past challenges and/or responses.

It is possible for intermediate BP nodes to encapsulate-and-encrypt Challenge and/or Response Bundles while they traverse untrusted networks, but that is a DTN configuration matter outside of the scope of this document.

7.2. Threat: BP Node Impersonation

As described in Section 8.1 of [RFC8555], it is possible for an active attacker to alter data on both ACME client channel and the DTN validation channel.

The primary mitigation is to delegate bundle integrity sourcing to a trusted routing node near, in the sense of bundle routing topology, to the bundle source node as defined in Section 4. This is functionally similar to DKIM signing of [RFC6376] and provides some level of bundle origination.

Another way to mitigate single-path on-path attacks is to attempt validation of the same Node ID from multiple sources or via multiple bundle routing paths, as defined in Section 3.5. It is not a trivial task to guarantee bundle routing though, so more advanced techniques such as onion routing (using bundle-in-bundle encapsulation [I-D.ietf-dtn-bibect]) could be employed.

7.3. Threat: Bundle Replay

It is possible for an on-path attacker to replay both Challenge Bundles or Response Bundles. Even in a properly-configured DTN it is possible that intermediate bundle routers to use multicast forwarding of a unicast-destination bundle.

Ultimately, the point of the ACME bundle exchange is to derive a Key Authorization and its cryptographic digest and communicate it back to the ACME server for validation, so the uniqueness of the Key Authorization directly determines the scope of replay validity. The uniqueness of each token-bundle to each challenge bundle ensures that the Key Authorization is unique to the challenge bundle. The uniqueness of each token-chal to the ACME challenge ensures that the Key Authorization is unique to that ACME challenge.

Having each bundle's primary block and payload block covered by a BIB from a trusted security source (see Section 4) ensures that a replayed bundle cannot be altered in the blocks used by ACME. All together, these properties mean that there is no degraded security caused by replay of either a Challenge Bundle or a Response Bundle even in the case where the primary or payload block is not covered by a BIB. The worst that can come of bundle replay is the waste of network resources as described in Section 7.4.

7.4. Threat: Denial of Service

The behaviors described in this section all amount to a potential denial-of-service to a BP agent.

A malicious entity can continually send Challenge Bundles to a BP agent. The victim BP agent can ignore Challenge Bundles which do not conform to the specific time interval and challenge token for which the ACME client has informed the BP agent that challenges are expected. The victim BP agent can require all Challenge Bundles to be BIB-signed to ensure authenticity of the challenge.

A malicious entity can continually send Response Bundles to a BP agent. The victim BP agent can ignore Response Bundles which do not conform to the specific time interval or Source Node ID or challenge token for an active Node ID validation.

Similar to other validation methods, an ACME server validating a DTN Node ID could be used as a denial of service amplifier. For this reason any ACME server can rate-limit validation activities for individual clients and individual certificate requests.

7.5. Inherited Security Considerations

Because this protocol relies on ACME for part of its operation, the security considerations of [RFC8555] apply to all ACME client--server exchanges during Node ID validation.

Because this protocol relies on BPv7 for part of its operation, the security considerations of [I-D.ietf-dtn-bpbis] and [I-D.ietf-dtn-bpsec] apply to all BP messaging during Node ID validation.

7.6. Out-of-Scope BP Agent Communication

Although messaging between an ACME client or ACME server an its associated BP agent are out-of-scope for this document, both of those channels are critical to Node ID validation security. Either channel can potentially leak data or provide attack vectors if not properly secured. These channels need to protect against spoofing of messaging in both directions to avoid interruption of normal validation sequencing and to prevent false validations from succeeding.

The ACME server and its BP agent exchange the outgoing token-chal, token-bundle, and Key Authorization digest but these values do not need to be confidential (they are also in plaintext over the BP channel).

Depending on implementation details, the ACME client might transmit the client account key thumbprint to its BP agent to allow computing the Key Authorization digest on the BP agent. If an ACME client does transmit its client account key thumbprint to a BP agent, it is

important that this data is kept confidential because it provides the binding of the client account key to the Node ID validation (as well as for all other types of ACME validation). Avoiding this transmission would require a full round-trip between BP agent and ACME client, which can be undesirable if the two are separated by a long-delay network.

8. IANA Considerations

This specification adds to the ACME registry and BP registry for this behavior.

8.1. ACME Identifier Types

Within the "Automated Certificate Management Environment (ACME) Protocol" registry [IANA-ACME], the following entry has been added to the "ACME Identifier Types" sub-registry.

Label	Reference
uri	This specification and [RFC3986]

Table 1

8.2. ACME Validation Methods

Within the "Automated Certificate Management Environment (ACME) Protocol" registry [IANA-ACME], the following entry has been added to the "ACME Validation Methods" sub-registry.

Label	Identifier Type	ACME	Reference
dtn-nodeid-01	uri	Y	This specification

Table 2

8.3. Bundle Administrative Record Types

Within the "Bundle Protocol" registry [IANA-BP], the following entries have been added to the "Bundle Administrative Record Types" sub-registry.

[NOTE to the RFC Editor: For [RFC5050] compatibility the AR-TBD value needs to be no larger than 15, but such compatibility is not needed. For BPbis the AR-TBD value needs to be no larger than 65535 as defined by [I-D.sipos-bpv7-admin-iana].]

Bundle Protocol Version	Value	Description	Reference
7	AR-TBD	ACME Node ID Validation	This specification

Table 3

9. Acknowledgments

This specification is based on DTN use cases related to PKIX certificate issuance.

The workflow and terminology of this validation method was originally copied from the work of Alexey Melnikov in [RFC8823].

10. References

10.1. Normative References

[FIPS180-4] National Institute of Standards and Technology, "Secure Hash Standard (SHS)", FIPS PUB 180-4, August 2015, <https://csrc.nist.gov/publications/detail/fips/180/4/final>.

[IANA-ACME] IANA, "Automated Certificate Management Environment (ACME) Protocol", <https://www.iana.org/assignments/acme/>.

[IANA-BP] IANA, "Bundle Protocol", <https://www.iana.org/assignments/bundle/>.

[IANA-SMI] IANA, "Structure of Management Information (SMI) Numbers", <https://www.iana.org/assignments/smi-numbers/>.

[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>.

[RFC2985] Nystrom, M. and B. Kaliski, "PKCS #9: Selected Object Classes and Attribute Types Version 2.0", RFC 2985, DOI 10.17487/RFC2985, November 2000, <https://www.rfc-editor.org/info/rfc2985>.

[RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, DOI 10.17487/RFC3552, July 2003, <https://www.rfc-editor.org/info/rfc3552>.

[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>.

[RFC4086] Eastlake 3rd, D., Schiller, J., and S. Crocker, "Randomness Requirements for Security", BCP 106, RFC 4086, DOI 10.17487/RFC4086, June 2005, <https://www.rfc-editor.org/info/rfc4086>.

[RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, DOI 10.17487/RFC4648, October 2006, <https://www.rfc-editor.org/info/rfc4648>.

[RFC4838] Cerf, V., Burleigh, S., Hooke, A., Torgerson, L., Durst, R., Scott, K., Fall, K., and H. Weiss, "Delay-Tolerant Networking Architecture", RFC 4838, DOI 10.17487/RFC4838, April 2007, <https://www.rfc-editor.org/info/rfc4838>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/info/rfc5280>.

[RFC7942] Sheffer, Y. and A. Farrel, "Improving Awareness of Running Code: The Implementation Status Section", BCP 205, RFC 7942, DOI 10.17487/RFC7942, July 2016, <https://www.rfc-editor.org/info/rfc7942>.

[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/info/rfc8174>.

[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/info/rfc8555>.

[RFC8610] Birkholz, H., Vigano, C., and C. Bormann, "Concise Data Definition Language (CDDL): A Notational Convention to Express Concise Binary Object Representation (CBOR) and JSON Data Structures", RFC 8610, DOI 10.17487/RFC8610, June 2019, <https://www.rfc-editor.org/info/rfc8610>.

[I-D.ietf-dtn-bpbis] Burleigh, S., Fall, K., and E. J. Birrane, "Bundle Protocol Version 7", Work in Progress, Internet-Draft, draft-ietf-dtn-bpbis-31, 25 January 2021, <https://datatracker.ietf.org/doc/html/draft-ietf-dtn-bpbis-31>.

[I-D.ietf-dtn-bpsec] III, E. J. B. and K. McKeever, "Bundle Protocol Security Specification", Work in Progress, Internet-Draft, draft-ietf-dtn-bpsec-27, 16 February 2021, <https://datatracker.ietf.org/doc/html/draft-ietf-dtn-bpsec-27>.

10.2. Informative References

[RFC5050] Scott, K. and S. Burleigh, "Bundle Protocol Specification", RFC 5050, DOI 10.17487/RFC5050, November 2007, <https://www.rfc-editor.org/info/rfc5050>.

[RFC6376] Crocker, D., Ed., Hansen, T., Ed., and M. Kucherawy, Ed., "DomainKeys Identified Mail (DKIM) Signatures", STD 76, RFC 6376, DOI 10.17487/RFC6376, September 2011, <https://www.rfc-editor.org/info/rfc6376>.

[RFC8823] Melnikov, A., "Extensions to Automatic Certificate Management Environment for End-User S/MIME Certificates", RFC 8823, DOI 10.17487/RFC8823, April 2021, <https://www.rfc-editor.org/info/rfc8823>.

[I-D.ietf-dtn-bibect] Burleigh, S., "Bundle-in-Bundle Encapsulation", Work in Progress, Internet-Draft, draft-ietf-dtn-bibect-03, 18 February 2020, <https://datatracker.ietf.org/doc/html/draft-ietf-dtn-bibect-03>.

[I-D.ietf-dtn-tcpclv4]
Sipos, B., Demmer, M., Ott, J., and S. Perreault, "Delay-Tolerant Networking TCP Convergence Layer Protocol Version 4", Work in Progress, Internet-Draft, draft-ietf-dtn-tcpclv4-28, 6 October 2021, <https://datatracker.ietf.org/doc/html/draft-ietf-dtn-tcpclv4-28>.

[I-D.sipos-dtn-udpcl]
Sipos, B., "Delay-Tolerant Networking UDP Convergence Layer Protocol", Work in Progress, Internet-Draft, draft-sipos-dtn-udpcl-01, 26 March 2021, <https://datatracker.ietf.org/doc/html/draft-sipos-dtn-udpcl-01>.

[I-D.sipos-bpv7-admin-iana]
Sipos, B., "Bundle Protocol Version 7 Administrative Record Types Registry", Work in Progress, Internet-Draft, draft-sipos-bpv7-admin-iana-00, 13 October 2021, <https://datatracker.ietf.org/doc/html/draft-sipos-bpv7-admin-iana-00>.

[I-D.bsipos-dtn-bpsec-cose]
Sipos, B., "DTN Bundle Protocol Security COSE Security Context", Work in Progress, Internet-Draft, draft-bsipos-dtn-bpsec-cose-06, 3 June 2021, <https://datatracker.ietf.org/doc/html/draft-bsipos-dtn-bpsec-cose-06>.

[github-dtn-demo-agent]
Sipos, B., "Python implementation of basic BPv7-related protocols", <https://github.com/BSipos-RKF/dtn-demo-agent/>.

[github-dtn-wireshark]
Sipos, B., "Wireshark Dissectors for BPv7-related Protocols", <https://github.com/BSipos-RKF/dtn-wireshark/>.

[LE-multi-perspective]
Aas, J., McCarney, D., and R. Shoemaker, "Multi-Perspective Validation Improves Domain Validation Security", 19 February 2020, <https://letsencrypt.org/2020/02/19/multi-perspective-validation.html>.

Appendix A. Administrative Record Types CDDL

[NOTE to the RFC Editor: The "0xFFFF" in this CDDL is replaced by the "ACME Node ID Validation" administrative record type code.]

The CDDL extension of BP [I-D.ietf-dtn-bpbis] for the ACME bundles is:

```
; All ACME records have the same structure
$admin-record /= [0xFFFF, acme-record]
acme-record = {
  token-chal,
  token-bundle,
  ? key-auth-digest ; present for Response Bundles
}
token-chal = (1 => bstr)
token-bundle = (2 => bstr)
key-auth-digest = (3 => bstr)
```

Appendix B. Example Authorization

[NOTE to the RFC Editor: The "0xFFFF" in these examples are replaced by the "ACME Node ID Validation" administrative record type code.]

This example is a bundle exchange for the ACME server with Node ID "dtn://acme-server/" performing a verification for ACME client Node ID "dtn://acme-client/". The example bundles use no block CRC or BPSec integrity, which is for simplicity and is not recommended for normal use. The provided figures are extended diagnostic notation [RFC8610].

For this example the ACME client key thumbprint has the base64url encoded value of:

"LPJNul-wow4m6DsqxbninhsWHlwfp0JecwQzYpOLmCQ"

And the ACME-server generated token-chal has the base64url-encoded value of:

"tPUZNY4ONIk6LxErRFEjVw"

B.1. Challenge Bundle

For the single challenge bundle in this example, the token-bundle (transported as byte string via BP) has the base64url-encoded value of:

"p3yRYFU4KxwQaHQjJ2RdiQ"

The minimal-but-valid Challenge Bundle is shown in Figure 2. This challenge requires that the ACME client respond within a 60 second time window.

```
[
  [
    7, / BP version /
    0x22, / flags: user-app-ack, payload-is-an-admin-record /
    0, / CRC type: none /
    [1, "//acme-client/"], / destination /
    [1, "//acme-server/"], / source /
    [1, "//acme-server/"], / report-to /
    [1000000, 0], / timestamp: 2000-01-01T00:16:40+00:00 /
    60000 / lifetime: 60s /
  ],
  [
    1, / block type code /
    1, / block number /
    0, / flags /
    0, / CRC type: none /
    <<[ / type-specific data /
      0xFFFF, / record-type-code /
      { / record-content /
        1: b64'tPUZNY4ONIk6LxErRFEjVw' / token-chal /
        2: b64'p3yRYFU4KxwQaHQjJ2RdiQ' / token-bundle /
      }
    ]>>
  ]
]
```

Figure 2: Example Challenge Bundle

B.2. Response Bundle

When the tokens are combined with the key thumbprint, the full Key Authorization value (a single string split across lines for readability) is:

```
"p3yRYFU4KxwQaHQjJ2RdiQtPUZNY4ONIk6LxErRFEjVw."
"LPJNul-wow4m6DsqxbninhsWHlwfp0JecwQzYpOLmCQ"
```

The minimal-but-valid Response Bundle is shown in Figure 3. This response indicates that there is 30 seconds remaining in the response time window.

```
[
  [
    7, / BP version /
    0x02, / flags: payload-is-an-admin-record /
    0, / CRC type: none /
    [1, "//acme-server/"], / destination /
    [1, "//acme-client/"], / source /
    [1, 0], / report-to: none /
    [1030000, 0], / timestamp: 2000-01-01T00:17:10+00:00 /
    30000 / lifetime: 30s /
  ],
  [
    1, / block type code /
    1, / block number /
    0, / flags /
    0, / CRC type: none /
    <<[ / block-type-specific data /
      0xFFFF, / record-type-code /
      { / record-content /
        1: b64'tPUZNY4ONIk6LxErRFEjVw' / token-chal /
        2: b64'p3yRYFU4KxwQaHQjJ2RdiQ' / token-bundle /
        3: b64'mVIOJEQZie8XpYM6MMVSQUiNPH64URnhM9niJ5XHrew'
        / key auth. digest /
      }
    ]>>
  ]
]
```

Figure 3: Example Response Bundle

Author's Address

Brian Sipos
RKF Engineering Solutions, LLC
7500 Old Georgetown Road
Suite 1275
Bethesda, MD 20814-6198
United States of America

Email: brian.sipos+ietf@gmail.com