

# ACME Persistent DNS Challenge Update

---

**draft-ietf-acme-dns-persist-00**

**Shiloh Heurich** (Fastly)

**Henry Birge-Lee** (Crosslayer Labs)

**Michael Slaughter** (Amazon Trust Services)

IETF ACME WG - Montreal

# Status Update

---

WG adopted October 16, 2025

draft-sheurich-acme-dns-persist → draft-ietf-acme-dns-persist

## Parallel Industry Progress:

- **Oct 9:** CA/BF SC088v3 **PASSED** (26 CAs YES, 3 consumers YES)
- **Nov 8:** IP Rights Review completes

## Implementation Status:

- **Committed (2026):** Fastly/Certainly, Let's Encrypt
- **Assessing:** Amazon Trust Services

## Proof of Concept:

pebble server, eggsampler client

- Full interoperability with regular and wildcard issuance

# How dns-persist-01 Works

---

## Record Format

```
_validation-persist.example.com. IN TXT  
"ca.example; accounturi=https://ca.example/acct/123"
```

## Key Features

- ✓ **Persistent** - Reuse for multiple certificates
- ✓ **Account-bound** - accounturi parameter
- ✓ **CA-specific** - Issuer domain name
- ✓ **Multi-CA** - Separate TXT records per issuer
- ✓ **Expiration** - Optional persistUntil
- ✓ **Scope** - policy=wildcard covers subdomains

# Why We Need This

---

## Current Challenges

### **http-01/tls-alpn-01**

- Port 80/443 required
- Geo-blocking blocks validation
- Wildcards unsupported

### **dns-01**

- DNS propagation delays
- API keys on servers risk compromise
- Automation complexity

### **dns-account-01**

- Scoped labels per account
- Addresses multi-region needs
- Still requires per-validation provisioning

## Current Workarounds

### **"Magic CNAMEs"** (acme-dns.io)

- Single point of failure
- DNS cache poisoning risk
- BGP hijacking vulnerability

# Why Standardize?

---

CAs could deploy via pre-validation without protocol changes, but this bypasses ACME's challenge selection.

**Standardization enables proper protocol integration.**

**Key insight:** ACME account URIs provide durable, cryptographic binding

# Design Principles

---

## Strong Account Binding

- **ACME accounturi:** Durable identity
- Prevents unauthorized use with DNS access alone
- Survives account key rotation
- No new trust anchors

## Multi-CA Architecture

- **Per-issuer TXT records**
- Each CA validates only their records
- No CA coordination required
- Domain owner controls authorization

## Flexibility & Extensibility

- **Optional parameters:**
  - `persistUntil` - explicit expiration
  - `policy=wildcard` - subdomain coverage
- **Unknown parameters ignored**
- Enables future protocol extensions

## Security Constraints

- **MUST respect DNS TTL**
- Bounded by CA policy limits
- DNSSEC SHOULD be validated

# Active WG Discussions

---

## Security Trade-offs

- Freshness vs. operational simplicity
- Account key becomes long-lived credential
- **Key compromise:** Immediate issuance without DNS access
- **Privacy:** `accounturi` in public DNS

## Validation Reuse Period

**Effective period = shortest of:**

- DNS record TTL (MUST respect)
- `persistUntil` parameter
- CA/BF policy (398d → 10d by 2029)

## DNSSEC Validation

- **Draft:** SHOULD validate signatures
- **Alternative:** MUST use validating resolver
- **Trade-off:** Security vs. private PKI flexibility

# Evolution to WG Draft

---

Changes from `draft-sheurich-acme-dns-persist-00` through WG adoption:

✓ **Pre-validation optimization**

- CA checks existing records during order creation
- Authorization becomes "valid" immediately if found
- Skips challenge flow with persistent record

✓ **Security Considerations expanded** - Record risks, account binding, subdomain validation

✓ **Long TXT record guidance** - Multi-string format for >255 characters

✓ **Error handling** - `malformed` for syntax, `unauthorized` for auth failures

✓ **Document renamed** - `draft-ietf-acme-dns-persist` (WG adoption)

# Seeking WG Input

---

**Acknowledging concerns:** Use case definition, trust relationships, existing validation interactions

## Questions:

### 1. DNSSEC requirement?

- Draft: SHOULD validate
- Alternative: MUST validate
- Which level?

### 2. Protocol validation caps?

Beyond `persistUntil`

### 3. Security considerations?

What else to address?

### 4. Timeline?

Given industry momentum

### 5. AccountURI flexibility? (PR #30)

Multiple URIs per account?

- **Pro:** Privacy, access control
- **Con:** CA cannot predict client's choice
- Trade-off: Privacy vs. simplicity

# Path Forward

---

- Incorporate Montreal feedback
- Expand security considerations
- Resolve PR #30 (accounturi flexibility)
- Address use case and trust concerns
- Target WGLC after 1-2 revisions
- Coordinate with CA/Browser Forum

**Feedback → Revision → WGLC → RFC**

# Questions & Discussion

---

**Thank you!**

**Contact:**

- Mailing list: [acme@ietf.org](mailto:acme@ietf.org)
- GitHub: <https://github.com/ietf-wg-acme/draft-ietf-acme-dns-persist>
- Draft: <https://datatracker.ietf.org/doc/draft-ietf-acme-dns-persist/>