IETF-100

ACME Token Identifier and Challenges

draft-barnes-acme-token-challenge-00
draft-barnes-acme-service-provider-code-00
draft-ietf-acme-service-provide-02

mbarnes@iconectiv.com

November 13, 2017

Overview

- Feedback @ IETF-99 suggested that a more generic token/ challenge mechanism could be used for Service Provider code token challenge (draft-ietf-acme-service-provider)
- Alternative proposal in draft-peterson-acme-authority-token (slightly different perspective)
- Minimal changes to existing WG document

Changes to draft-ietf-acme-service-provider-02

- Added text about the lifetime of the service provider code token
- Changed “sub” field in JWT token to be a string and not an array of strings.

*

draft-barnes-acme-token-challenge

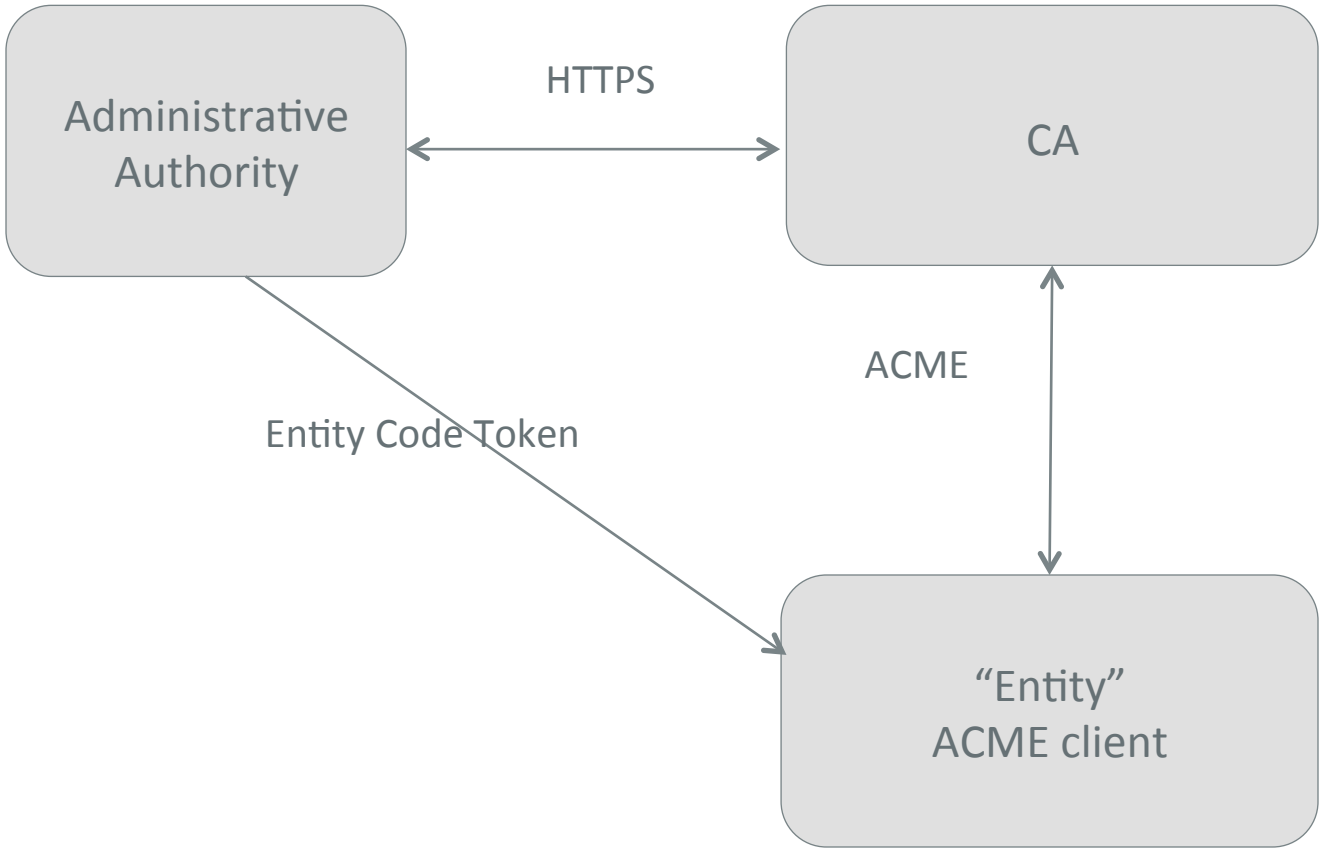
- Mechanism effectively the same as draft-acme-service-provider:
  - Rather than a Service Provider Code, a more generic name is assigned (“entityCode”*).
  - Acquisition mechanism and validation mechanism follows the same control flow.
  - The entity requesting a certificate has a relationship with an administrative authority which assigns a unique code to the entity.
  - The token for the challenge response is issued by the administrative authority with whom the Certification Authority (CA) also has a trust relationship.
    - The entity code is included as part of the token that the administrative authority issues.

* Other terms considered: “serviceCode” or “authCode”

draft-barnes-acme-service-provider-code

- Defines the specific usage of the mechanism defined in draft-barnes-acme-token-challenge to support Service Provider codes
- If generic mechanism progresses, this document is starting point for updates required for draft-ietf-acme-service-provider

Architecture for token challenge



Entity Code Token

JWT Header:

- alg: Defines the algorithm used in the signature of the token. *For Service Provider Code tokens, the algorithm MUST be "ES256".*
- typ: Set to standard "JWT" value.
- x5u: Defines the URL of the certificate of the ~~*STI-PA*~~ *Administrative Authority* validating the token.

JWT Payload:

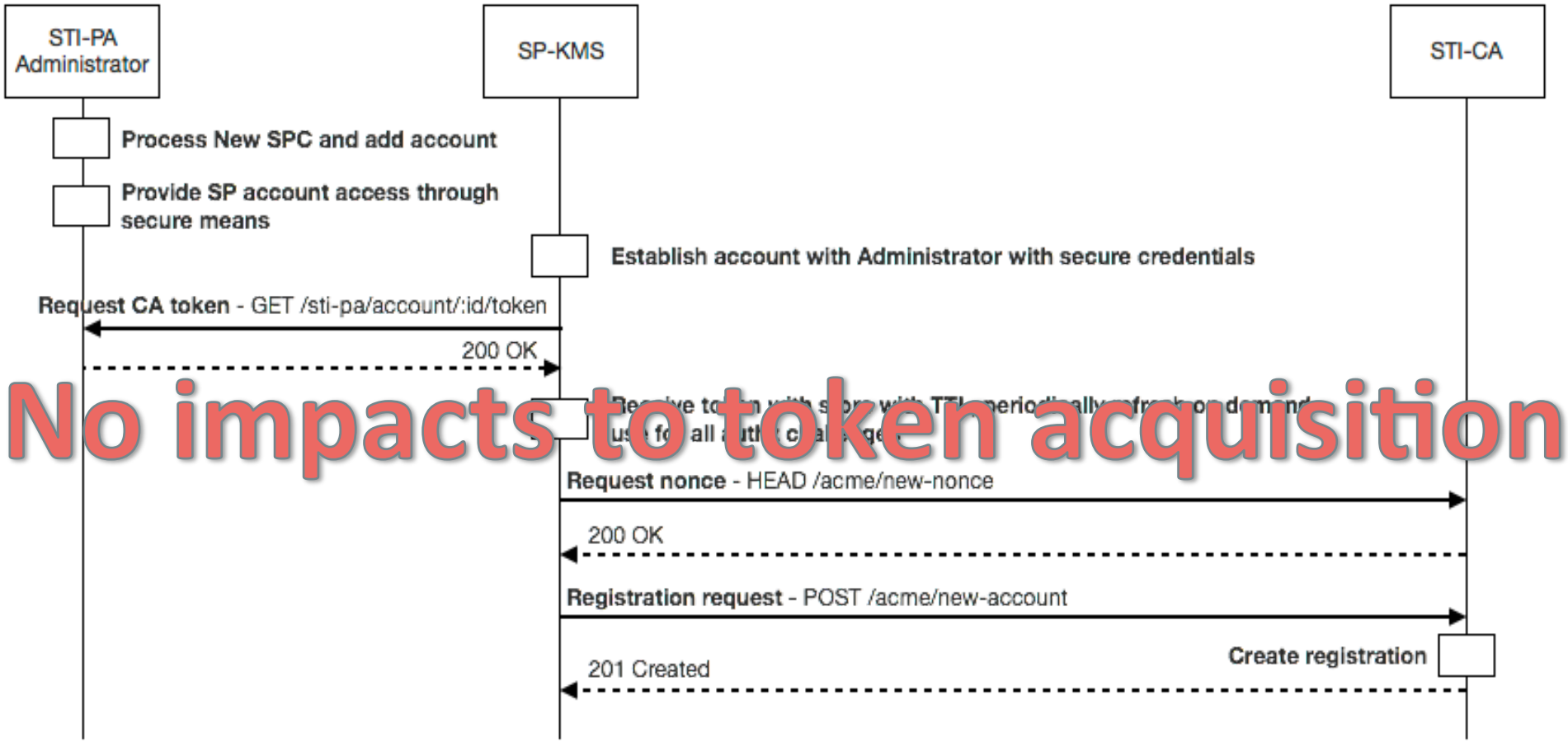
- *sub (*) Entity code token value being validated in the form of an ASCII string.*
- iat: DateTime value of the time and date the token was issued.
- nbf: DateTime value of the starting time and date that the token is valid.
- exp: DateTime value of the ending time and date that the token expires.
- fingerprint: : (Certificate) key fingerprint of the ACME credentials the *Entity* used to create an account with the CA.

"fingerprint" is of the form:

base64url(JWK_Thumbprint(accountKey))

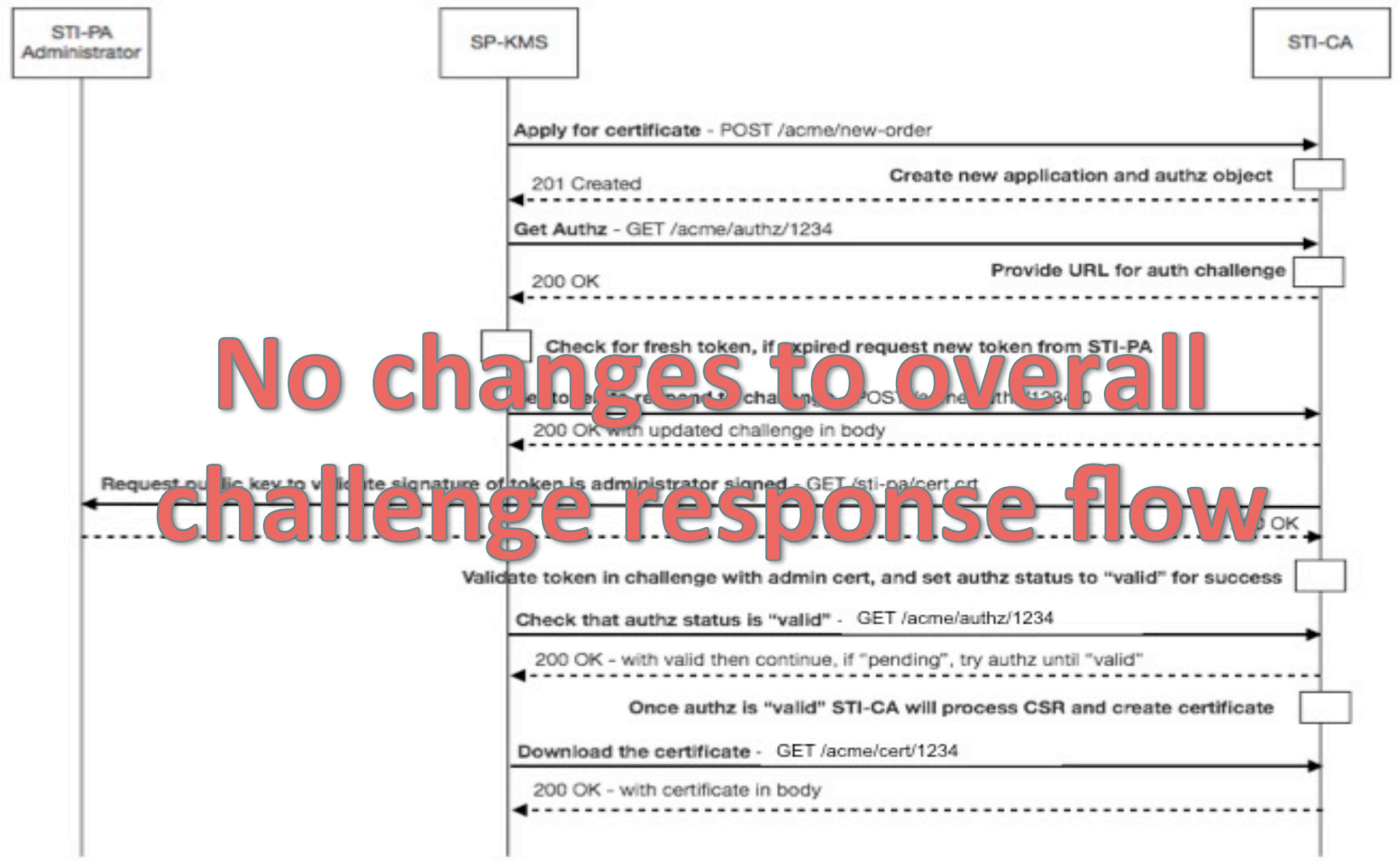
* Changed from array of strings to a single string (sufficient for ATIS-1000080)

STI-PA Account Setup, SPC Token Acquisition, ACME Acct Registration



No impacts to token acquisition

Certificate Acquisition



No changes to overall challenge response flow

Discussion points

1. Identifier defined in draft-peterson-acme-authority-token introduces a slightly different model:
   - Token relates to authority and not specific entity/service provider to whom code/token are assigned.
   - An authority would assign unique tokens to unique entities for which it has assigned a unique identifier.
2. STIR TNAuthList includes both TNs and Service Provider Codes
   - Service Provider codes are significantly different in structure and use than TNs
3. Challenge type is no longer specific to Service Provider Codes
   - Fairly simple approach but genericity requires consideration of other practical use cases prior to publication
     - Could could slow down progression of this document (implementations already done and underway using service provider code)