SCIM Credential Mgmt

IETF91

Phil Hunt, Oracle

November 14, 2014

Credential Management

- Users for SPs using SCIM as main RESTful Identity API
  - Used in provisioning and directory modes
  - Customers building or acquiring self-service suites have been requesting a "standard" methodology for password mgmt
- Devices/Apps (e.g. OAuth)
  - Negotiating proof of possession tokens, symmetric/asym keys and other credential

Interop Requirement

- Customers would like to integrate independently developed user-interface components with SCIM Service Provider features based on a standard.
- Device credential management is likely an upcoming issue (e.g. OAuth POP tokens)

Password Management

- Passwords Mgmt is often a "process"
  - Doesn't fit well as an "attribute" modify
  - Complex policy causes multiple error conditions
  - Multiple resources and attributes checked
  - Multiple attributes affected by a password change
    - Password, Password History, etc
- Password recovery involves unauthenticated requests
  - Requests may be long-running

RESTful Requests as SCIM Extensions

- Model multi-component and multi-step processes as stateful "requests"
- For each request type, define a new SCIM ResourceType
  - /PasswordValidateRequest
  - /PasswordResetWithChallengesRequest
  - others:
    - UserName validation / generation / recovery

Request Processing

- Requests may involve secondary / off-line verification
  - e-mail confirmation
- Requests may come from
  - Trusted/authenticated web UI client
  - Mobile application
  - Javascript
- Requests only persist while transaction outstanding (typically no GET)
- There are security considerations (e.g. DoS)

Password Reset W/Challenges

```
POST /PasswordResetterWithChallengesRequest HTTP/1.1 Host:
example.com
Authorization: Bearer h480djs93hd8
{
"schemas":["urn:ietf:params:scim:schema:oracle:core:
2.0:PasswordResetterWithChallengesRequest"],
"$ref": "/Users/2819c223-7f76-453a-919d-41
"challenges": [
            { "challenge":"what is your favorite color",
              "response":"color" },
            { "challenge":"what is name of
              "response":"pet"},
            { "challenge":"what is city of your birth",
              "response":"city"}
        ],
"password": "<new password>"
}
```

May be an OAuth Client

Request Object Schema

The target of the request

Password Validation

```
POST /PasswordValidateRequest  HTTP/1.1
Host: example.com
Accept: application/json
Content-Type: application/json
Content-Length: ...
{
 "schemas":["urn:ietf:params:scim:schema:oracle:
   core:2.0:PasswordValidateRequest"],
 "$ref": “/Users/
2819c223-7f76-453a-919d-413861904646”,
 "password":"<passwordValue>"
}
```

Response:

```
HTTP/1.1 200 OK
```

Discussion

- Is credential management something we want to work on?

- Should we do Password Mgmt as a separate extension?