

OpenAPI SCIM 2.0 documentation

SCIM 2.0 endpoints are available now under for OpenAPI integrations. Clients can now provision users directly from their ID provider.

ID Providers Supported:

1. Microsoft Azure

Attributes mapping and support:

Scope Attribute	SCIM Attribute	
email	userName	This attribute is treated as email in scope and is unique identified. SCIM <code>emails</code> will be ignored.
firstName	name.givenName	
lastName	name.lastName	
enabled	active	true or false

API endpoints:

1. Find Users:

API Endpoint: /scim/v2/Users/<User_ID>?filter=<query>

Method: Get

Query Params:

1. filter: Only `userName` is supported as filter along with `eq` comparison operator.

```
userName eq "test@example.com"
```

Example Request:

```
1 GET /v1.0/scim/v2/Users?filter=userName+eq+%22testexample.com%22 HTTP/1.1
2 Host: 87dd-122-162-190-203.in.ngrok.io
```

Example Response:

```
1 {
2   "schemas": [
3     "urn:ietf:params:scim:api:messages:2.0:ListResponse"
4   ],
5   "totalResults": 1,
6   "startIndex": 1,
7   "itemsPerPage": 9999,
8   "resources": [
```

```

9      {
10         "schemas": [
11             "urn:ietf:params:scim:schemas:core:2.0:User"
12         ],
13         "id": "247",
14         "userName": "bradpitt@azure.com",
15         "name": {
16             "givenName": "brad",
17             "familyName": "pitt"
18         },
19         "emails": [
20             {
21                 "primary": true,
22                 "value": "bradpitt@azure.com",
23                 "type": null
24             }
25         ],
26         "externalId": null,
27         "active": false,
28         "groups": []
29     }
30 ]
31 }

```

2. Create User

API Endpoint: /scim/v2/Users

Method: POST

Body: Standard SCIM User Resource

```

1  {
2      "schemas": [
3          "urn:ietf:params:scim:schemas:core:2.0:User",
4          "urn:ietf:params:scim:schemas:extension:enterprise:2.0:User"
5      ],
6      "externalId": "user20",
7      "userName": "user20@dbalani359gmail.onmicrosoft.com",
8      "active": true,
9      "displayName": "User 20",
10     "meta": {
11         "resourceType": "User"
12     },
13     "name": {
14         "formatted": "User 20",
15         "familyName": "20",
16         "givenName": "User"
17     }
18 }

```

Output: Standard SCIM User Object with unique ID

```

1  {
2      "schemas": [

```

```
3      "urn:ietf:params:scim:schemas:core:2.0:User"
4    ],
5    "id": "251",
6    "userName": "user20@dbalani359gmail.onmicrosoft.com",
7    "name": {
8      "givenName": "User",
9      "familyName": "20"
10   },
11   "emails": [
12     {
13       "primary": true,
14       "value": "user20@dbalani359gmail.onmicrosoft.com",
15       "type": null
16     }
17   ],
18   "externalId": null,
19   "active": true,
20   "groups": [],
21   "roles": [
22     "MEMBER"
23   ]
24 }
```

3. Find User

API Endpoint: /scim/v2/Users/<user_id>

Method: Get

Output: Standard scim resource response

```
1 {
2   "schemas": [
3     "urn:ietf:params:scim:schemas:core:2.0:User"
4   ],
5   "id": "251",
6   "userName": "user20@dbalani359gmail.onmicrosoft.com",
7   "name": {
8     "givenName": "User",
9     "familyName": "20"
10  },
11  "emails": [
12    {
13      "primary": true,
14      "value": "user20@dbalani359gmail.onmicrosoft.com",
15      "type": null
16    }
17  ],
18  "externalId": null,
19  "active": true,
20  "groups": [],
21  "roles": [
22    "MEMBER"
23  ]
24 }
```

4. Patch User

API: /scim/v2/Users

Method: Patch

Input: SCIM Patch

```
1 {
2   "schemas": [
3     "urn:ietf:params:scim:api:messages:2.0:PatchOp"
4   ],
5   "Operations": [
6     {
7       "op": "Replace",
8       "path": "active",
9       "value": "False"
10    },
11    {
12      "op": "Add",
13      "path": "displayName",
14      "value": "User 20."
15    },
16    {
17      "op": "Replace",
18      "path": "name.givenName",
19      "value": "User."
20    },
21    {
22      "op": "Replace",
23      "path": "name.familyName",
24      "value": "20."
25    },
26    {
27      "op": "Add",
28      "path": "name.formatted",
29      "value": "User. 20."
30    },
31    {
32      "op": "Add",
33      "path": "externalId",
34      "value": "user20"
35    }
36  ]
37 }
```

Note: Only attributes given in the table above will be considered others will be ignored.

OutPut: Updated Scim User:

```
1 {
2   "schemas": [
3     "urn:ietf:params:scim:schemas:core:2.0:User"
4   ],
5   "id": "251",
6   "userName": "user20@dbalani359gmail.onmicrosoft.com",
7   "name": {
```

```

8      "givenName": "User.",
9      "familyName": "20."
10   },
11   "emails": [
12     {
13       "primary": true,
14       "value": "user20@dbalani359gmail.onmicrosoft.com",
15       "type": null
16     }
17   ],
18   "externalId": null,
19   "active": false,
20   "groups": [],
21   "roles": [
22     "MEMBER"
23   ]
24 }

```

5. Put User:

API: /scim/v2/Users/<UserUD>

Method: Put

Request: Standard SCIM User Resource, you can omit the parts that are not changed, only attributes in tables are supported.

```

1 {
2   "schemas": [
3     "urn:ietf:params:scim:schemas:core:2.0:User"
4   ],
5   "id": "247",
6   "userName": "bradpitt.5@azure.com",
7   "active": false,
8   "groups": []
9 }

```

Response:

Updates SCIM User.

```

1 {
2   "schemas": [
3     "urn:ietf:params:scim:schemas:core:2.0:User"
4   ],
5   "id": "251",
6   "userName": "bradpitt.5@azure.com",
7   "name": {
8     "givenName": "User.",
9     "familyName": "20."
10  },
11  "emails": [
12    {
13      "primary": true,
14      "value": "bradpitt.5@azure.com",
15      "type": null
16    }
17  ],

```

```
18     "externalId": null,  
19     "active": false,  
20     "groups": [],  
21     "roles": [  
22         "MEMBER"  
23     ]  
24 }
```