



# Vermilion Reporting Suite (VRS) SCIM API Version 1.0

## Developer's Manual & Reference

Feb 2022

## VRS SCIM API – Version 1.0

### Contents

|  |          |
|--|----------|
| <b>VRS SCIM API – Version 1.0</b> .....              | <b>2</b> |
| 1. API Program .....                                 | 3        |
| 2. Vermilion Reporting Suite (VRS) API .....         | 4        |
| 3. SCIM .....  | 4        |
| 3.1. Retrieve VRS users.....                         | 5        |
| 3.2. Creates a user.....                             | 7        |
| 3.3. Retrieves a VRS user.....                       | 9        |
| 3.4. Deletes a VRS user .....                        | 11       |
| 3.5. Add or remove VRS user attributes.....          | 12       |
| 3.6. Updates a VRS user .....                        | 14       |
| 3.7. Retrieve a list of VRS roles/groups .....       | 16       |
| 3.8. Creates a VRS role/group .....                  | 18       |
| 3.9. Retrieves a VRS role/group.....                 | 19       |
| 3.10. Deletes a VRS role/group.....                  | 21       |
| 3.11. Add or remove a user to/from a role/group..... | 22       |
| 3.12. Updates a VRS role .....                       | 24       |
| 4. Troubleshooting.....                              | 26       |
| 4.1. Support – FactSet’s Issue Tracker .....         | 26       |
| 4.2. Submitting an Issue via Issue Tracker.....      | 26       |
| 4.3. Support Products .....                          | 27       |
| 5. Version Upgrade.....                              | 27       |

## 1. API Program

The aim of this API program is to enable our customers to link to VRS via a faster mechanism, using industry standard APIs and a similar API framework to other FactSet products, such as Publisher. The API provides more freedom of VRS functionality to our clients and enable them to build more custom solutions as well as a more sophisticated channel to interact with FactSet's market leading reporting solutions. As the market continues to demand more transparency and data, FactSet will provide flexible options to meet those demands. APIs complement the current reporting suite offerings and facilitate partnerships by enabling users to build private experiences.

The first stage of exposing VRS APIs will focus on the Report retrieval service and the Get data service areas of the system. Later, the program will expand to include other areas of the Vermilion platform, such as the Workflow service.

The Program will provide the following:

- Developer toolkit to build proof of concept
- Uniform feel across all FactSet's Enterprise scale APIs
- Adherence to industry standards
- Versioned APIs
- Documentation and tutorials on the developer portal.

## 2. Vermilion Reporting Suite (VRS) API

Through the Vermilion Reporting Suite (VRS) SCIM API, users can manage their users and groups (roles) using the SCIM standard.

All APIs are hosted under <https://api.factset.com>. Authentication is handled using API Keys and authorization is handled using FactSet's in-house subscriptions product. You can find more information about using API Keys at <https://developer.factset.com/authentication>.

HTTP request and response header names should be considered case insensitive as per [HTTP Standard](#). Please do not rely on case sensitive matching of headers in your code.

## 3. SCIM

From VRS version 21.9, our new users and roles API utilises System for Cross-domain Identity Management (SCIM) 2.0 standard. In order for clients to make use of the SCIM API, WildFly requires an update. Should you wish to benefit from this feature, please contact the Vermilion Support team for assistance.

### 3.1. Retrieve VRS users

/scim/v2/Users/

#### Description

SCIM: retrieves the full list of VRS users.

#### Request Parameters

| Parameter Name | Data type | Required | Description                                      |
|----------------|-----------|----------|--|
| count          | Integer   | No       | Non-negative maximum number of entries to return |
| startIndex     | Integer   | No       | The 1-based index of the first query result      |

#### Returns

| HTTP status code | Description                                 |
|------------------|---|
| 200              | Successful operation.                       |
| 401              | Unauthorised.                               |
| 403              | No permissions to view data sources in VRS. |

#### Example

**Request:**

GET <https://api.factset.com/scim/v2/Users?count=5&startIndex=0>

**Response:**

HTTP 200 Successful operation

**Body:**

```
{
  "schemas": [
    "urn:ietf:params:scim:api:messages:2.0:ListResponse"
  ],
  "totalResults": 5,
  "itemsPerPage": 5,
  "startIndex": 1,
  "Resources": [
    {
      "schemas": [
        "urn:scim:schemas:extension:FactSet:VRS:1.0:User"
      ],
      "id": "444",
      "userName": "testuser",
      "name": {
        "formatted": "testuser"
      }
    }
  ],
}
```

```
"userType": "0",
"active": true,
"authenticatedUserName": "",
"tenant": "TRAINING",
"emails": [
  {
    "value": "Training@factset.com",
    "type": "work",
    "primary": true
  }
],
"phoneNumbers": [
  {
    "value": "",
    "type": "work",
    "primary": true
  }
],
"domainCode": "API-FactSet",
"isAdministrator": false
},
]
```

### 3.2. Creates a user

/scim/v2/Users/

#### Description

SCIM: Creates a VRS user.

#### Request Parameters

N/A

#### Returns

| HTTP status code | Description   |
|------------------|---|
| 201              | Created   |
| 400              | Bad Request, required values not provided in JSON request body (e.g. tenant, name or userName (loginID))      |
| 403              | Forbidden, user does not have permission to create users or it does not have requested tenancy assigned to it |
| 409              | User already exists   |

#### Example

**Request:**

POST <https://api.factset.com/scim/v2/Users>

**Response:**

HTTP 201 Created

**Body:**

```
{
  "schemas": [
    "urn:scim:schemas:extension:FactSet:VRS:1.0:User"
  ],
  "id": "641",
  "userName": "SCIM_TEST",
  "name": {
    "formatted": "New User"
  },
  "userType": "0",
  "active": true,
  "authenticatedUserName": "",
  "tenant": "MASTER",
  "emails": [
    {
      "value": "abcdef@factset.com",
      "type": "work",
    }
  ]
}
```

```
        "primary": true
    }
],
"phoneNumbers": [
    {
        "value": "123",
        "type": "work",
        "primary": true
    }
],
"photos": [
    {
        "value": " https://api.factset.com/profilepicpath"
    }
],
"domainCode": "API-FactSet",
"isAdministrator": false,
"meta": {
    "resourceType": "VRSUser",
    "created": "2022-01-25T11:41:28.047Z",
    "lastModified": "2022-01-25T11:41:28.047Z",
    "location": " https://api.factset.com/scim/v2/Users/641"
}
}
```



### 3.3. Retrieves a VRS user

/scim/v2/Users/{userId}

#### Description

SCIM: Retrieves a VRS user based on their ID.

#### Request Parameters

| Parameter Name | Data type | Required | Description        |
|----------------|-----------|----------|--------------------|
| userId         | String    | Yes      | The ID of the user |

#### Returns

| HTTP status code | Description                                 |
|------------------|---|
| 200              | Successful operation.                       |
| 403              | No permissions to view data sources in VRS. |
| 404              | Not Found                                   |

#### Example

**Request:**

GET <https://api.factset.com/scim/v2/Users/321>

**Response:**

HTTP 200 Successful operation

**Body:**

```
{
  "schemas": [
    "urn:scim:schemas:extension:FactSet:VRS:1.0:User"
  ],
  "id": "321",
  "userName": "$!",
  "name": {
    "formatted": "$!"
  },
  "userType": "0",
  "active": true,
  "authenticatedUserName": "",
  "tenant": "QA",
  "emails": [
    {
      "value": "training@factset.com",
      "type": "work",
      "primary": true
    }
  ]
}
```

```
],
  "phoneNumbers": [
    {
      "value": "",
      "type": "work",
      "primary": true
    }
  ],
  "domainCode": "API-FactSet",
  "isAdministrator": false,
  "meta": {
    "resourceType": "VRSUser",
    "created": "2019-08-28T20:57:06.670Z",
    "lastModified": "2019-08-28T20:57:06.670Z",
    "location": " https://api.factset.com/scim/v2/Users/321"
  }
}
```

### 3.4. Deletes a VRS user

/scim/v2/Users/{userId}

#### Description

SCIM: Deletes a VRS user (this marks them as 'Deleted' in the database)

#### Request Parameters

| Parameter Name | Data type | Required | Description        |
|----------------|-----------|----------|--------------------|
| userId         | String    | Yes      | The ID of the user |

#### Returns

| HTTP status code | Description  |
|------------------|--|
| 204              | No content   |
| 403              | Forbidden, user does not have permission to delete users |
| 404              | Not Found  |

#### Example

**Request:**

DELETE <https://api.factset.com/scim/v2/Users/270>

**Response:**

HTTP 204 No Content

**Body:**

1

### 3.5. Add or remove VRS user attributes

/scim/v2/Users/{userId}

#### Description

SCIM: Adds or removes VRS user attributes based on the JSON properties

#### Request Parameters

| Parameter Name | Data type | Required | Description        |
|----------------|-----------|----------|--------------------|
| userId         | String    | Yes      | The ID of the user |

#### Returns

| HTTP status code | Description  |
|------------------|--|
| 200              | Successfully updated user                                |
| 403              | Forbidden, user does not have permission to update users |
| 404              | Not Found  |

#### Example

**Request:**

PATCH <https://api.factset.com/scim/v2/Users/628>

**Response:**

HTTP 200 Successful operation

**Body:**

```
{
  "schemas": [
    "urn:scim:schemas:extension:FactSet:VRS:1.0:User"
  ],
  "id": "640",
  "userName": "New_User01",
  "name": {
    "formatted": "New User 9401"
  },
  "userType": "0",
  "active": true,
  "authenticatedUserName": "SCIM_Patchuser",
  "tenant": "MASTER",
  "emails": [
    {
      "value": "abcdef@factset.com",
      "type": "work",
      "primary": true
    }
  ]
}
```

```
],
  "phoneNumbers": [
    {
      "value": "123",
      "type": "work",
      "primary": true
    }
  ],
  "photos": [
    {
      "value": " https://api.factset.com/profilepicpath"
    }
  ],
  "domainCode": "API-Factset",
  "isAdministrator": false,
  "meta": {
    "resourceType": "VRSUser",
    "created": "2022-01-25T11:40:11.717Z",
    "lastModified": "2022-01-25T06:46:37.9379092-05:00",
    "location": " https://api.factset.com/scim/v2/Users/640"
  }
}
```

### 3.6. Updates a VRS user

/scim/v2/Users/{userId}

#### Description

SCIM: Updates (replaces) a VRS user with the properties in the JSON

#### Request Parameters

| Parameter Name | Data type | Required | Description        |
|----------------|-----------|----------|--------------------|
| userId         | String    | Yes      | The ID of the user |

#### Returns

| HTTP status code | Description  |
|------------------|--|
| 200              | Successfully updated user                                |
| 403              | Forbidden, user does not have permission to update users |
| 404              | Not Found  |

#### Example

```

Request:
PUT https://api.factset.com/scim/v2/Users/10

Response:
HTTP 200 Successful operation

Body:
{
  "schemas": [
    "urn:scim:schemas:extension:FactSet:VRS:1.0:User"
  ],
  "id": "640",
  "userName": "API tests",
  "name": {
    "formatted": "API Test Account"
  },
  "userType": "0",
  "active": true,
  "authenticatedUserName": "",
  "tenant": "MASTER",
  "emails": [
    {
      "value": "training@factset.com",
      "type": "work",
      "primary": true
    }
  ]
}
    
```

```
],  
  "phoneNumbers": [  
    {  
      "value": "12349000",  
      "type": "work",  
      "primary": true  
    }  
  ],  
  "domainCode": "API-FactSet",  
  "isAdministrator": false,  
  "meta": {  
    "resourceType": "VRSUser",  
    "location": " https://api.factset.com/scim/v2/Users/640"  
  }  
}
```

### 3.7. Retrieve a list of VRS roles/groups

/scim/v2/Groups/

#### Description

SCIM: Retrieves a VRS roles/groups

#### Request Parameters

| Parameter Name | Data type | Required | Description                                      |
|----------------|-----------|----------|--|
| count          | Integer   | No       | Non-negative maximum number of entries to return |
| startIndex     | Integer   | No       | The 1-based index of the first query result      |

#### Returns

| HTTP status code | Description   |
|------------------|---|
| 200              | Successful operation  |
| 403              | Forbidden, user does not have permission to view roles/groups |

#### Example

**Request:**

GET <https://api.factset.com/scim/v2/Groups?count=5&startIndex=2>

**Response:**

HTTP 200 Successful operation

**Body:**

```
{
  "schemas": [
    "urn:ietf:params:scim:api:messages:2.0:ListResponse"
  ],
  "totalResults": 5,
  "itemsPerPage": 4,
  "startIndex": 2,
  "Resources": [
    {
      "schemas": [
        "urn:scim:schemas:extension:FactSet:VRS:1.0:Group"
      ],
      "id": "176",
      "displayName": "User 1",
      "members": [
        {
          "value": "201",
```



```
        "$ref": " https://api.factset.com/scim/v2/Users/201",
        "type": "User"
    },
    {
        "value": "264",
        "$ref": " https://api.factset.com/scim/v2/Users/264",
        "type": "User"
    },
    {
        "value": "289",
        "$ref": " https://api.factset.com/scim/v2/Users/289",
        "type": "User"
    },
    {
        "value": "291",
        "$ref": " https://api.factset.com/scim/v2/Users/291",
        "type": "User"
    },
    {
        "value": "623",
        "$ref": " https://api.factset.com/scim/v2/Users/623",
        "type": "User"
    }
],
"tenant": "MASTER",
"domainCode": "API-FactSet"
}
]
```

## 3.8. Creates a VRS role/group

/scim/v2/Groups/

**Description**

SCIM: Creates a VRS role/group

**Request Parameters**

N/A

**Returns**

| HTTP status code | Description   |
|------------------|---|
| 201              | Created   |
| 403              | Forbidden, user does not have permission to create roles/groups |
| 409              | Role/Group already exists                                       |

**Example****Request:**POST <https://api.factset.com/scim/v2/Groups>**Response:**

HTTP 201 Created

**Body:**

```
{
  "schemas": [
    "urn:scim:schemas:extension:FactSet:VRS:1.0:Group"
  ],
  "id": "2205",
  "displayName": "test_scim_role",
  "tenant": "MASTER",
  "domainCode": "API-FactSet",
  "meta": {
    "resourceType": "VRSGroup",
    "created": "2022-01-25T11:52:53.567Z",
    "lastModified": "2022-01-25T11:52:53.567Z",
    "location": " https://api.factset.com/scim/v2/Groups/2205"
  }
}
```

### 3.9. Retrieves a VRS role/group

/scim/v2/Groups/{groupId}

#### Description

SCIM: Retrieves a VRS role/group based on the ID

#### Request Parameters

| Parameter Name | Data type | Required | Description              |
|----------------|-----------|----------|--------------------------|
| groupId        | String    | Yes      | The ID of the role/group |

#### Returns

| HTTP status code | Description   |
|------------------|---|
| 200              | Successful operation  |
| 403              | Forbidden, user does not have permission to view roles/groups |
| 404              | Not Found   |

#### Example

```

Request:
GET https://api.factset.com/scim/v2/Groups/1

Response:
HTTP 200 Successful operation

Body:
{
  "schemas": [
    "urn:scim:schemas:extension:FactSet:VRS:1.0:Group"
  ],
  "id": "1",
  "displayName": "Admin",
  "members": [
    {
      "value": "10",
      "$ref": "https://api.factset.com/scim/v2/Users/10",
      "type": "User"
    },
    {
      "value": "11",
      "$ref": "https://api.factset.com/scim/v2/Users/11",
      "type": "User"
    },
    {
      "value": "14",

```

```
    "$ref": "https://api.factset.com/scim/v2/Users/14",
    "type": "User"
  },
  {
    "value": "15",
    "$ref": "https://api.factset.com/scim/v2/Users/15",
    "type": "User"
  },
  {
    "value": "26",
    "$ref": "https://api.factset.com/scim/v2/Users/26",
    "type": "User"
  },
  {
    "value": "623",
    "$ref": "https://api.factset.com/scim/v2/Users/623",
    "type": "User"
  }
],
"tenant": "MASTER",
"domainCode": "API-FactSet",
"meta": {
  "resourceType": "VRSGroup",
  "created": "2020-08-25T13:50:10.607Z",
  "lastModified": "2020-08-25T13:50:10.607Z",
  "location": "https://api.factset.com/scim/v2/Groups/1"
}
}
```

### 3.10. Deletes a VRS role/group

/scim/v2/Groups/{groupId}

#### Description

SCIM: Deletes a VRS role/group

#### Request Parameters

| Parameter Name | Data type | Required | Description              |
|----------------|-----------|----------|--------------------------|
| groupId        | String    | Yes      | The ID of the role/group |

#### Returns

| HTTP status code | Description   |
|------------------|---|
| 204              | No content  |
| 403              | Forbidden, user does not have permission to delete roles/groups |
| 404              | Not Found   |

#### Example

**Request:**

DELETE <https://api.factset.com/scim/v2/Groups/2205>

**Response:**

HTTP 204 No content

**Body:**

1

### 3.11. Add or remove a user to/from a role/group

/scim/v2/Groups/{groupId}

#### Description

SCIM: Adds or removes a VRS user to/from a VRS role/group

#### Request Parameters

| Parameter Name | Data type | Required | Description              |
|----------------|-----------|----------|--------------------------|
| groupId        | String    | Yes      | The ID of the role/group |

#### Returns

| HTTP status code | Description                           |
|------------------|---------------------------------------|
| 204              | Successfully added user to role/group |

#### Example

**Request:**

PATCH <https://api.factset.com/scim/v2/Groups/2205>

**Response:**

HTTP 200 Successful operation

**Body:**

```
{
  "schemas": [
    "urn:scim:schemas:extension:FactSet:VRS:1.0:Group"
  ],
  "id": "2205",
  "displayName": "test_scim_role",
  "members": [
    {
      "value": "11",
      "$ref": "https://api.factset.com/scim/v2/Users/11",
      "type": "User"
    }
  ],
  "tenant": "MASTER",
  "domainCode": "API-FactSet",
  "meta": {
    "resourceType": "VRSGroup",
    "created": "2022-01-25T11:52:53.567Z",
    "lastModified": "2022-01-25T11:52:53.567Z",
    "location": "https://api.factset.com/scim/v2/Groups/2205"
  }
}
```

|   |
|---|
| } |
|---|

### 3.12. Updates a VRS role

/scim/v2/Groups/{groupId}

#### Description

SCIM: Updates (replaces) a VRS role/group with the properties in the JSON

#### Request Parameters

| Parameter Name | Data type | Required | Description              |
|----------------|-----------|----------|--------------------------|
| groupId        | String    | Yes      | The ID of the role/group |

#### Returns

| HTTP status code | Description   |
|------------------|---|
| 200              | Successfully updated role/group                                 |
| 403              | Forbidden, user does not have permission to update roles/groups |
| 404              | Not Found   |

#### Example

```

Request:
PUT https://api.factset.com/scim/v2/Groups/623

Response:
HTTP 200 Successful operation

Body:
{
  "schemas": [
    "urn:scim:schemas:extension:FactSet:VRS:1.0:Group"
  ],
  "id": "623",
  "displayName": "API_auto_03_03",
  "members": [
    {
      "value": "354",
      "$ref": "https://api.factset.com/scim/v2/Users/354",
      "type": "User"
    },
    {
      "value": "623",
      "$ref": "https://api.factset.com/scim/v2/Users/623",
      "type": "User"
    }
  ],
  "tenant": "TESTIMPORT",

```



```
"domainCode": "API-FactSet",  
"meta": {  
  "resourceType": "VRSGroup",  
  "created": "2022-01-21T15:56:37.783Z",  
  "lastModified": "2022-01-21T15:56:37.783Z",  
  "location": "https://api.factset.com/scim/v2/Groups/623"  
}  
}
```

## 4. Troubleshooting

The following steps are recommended to troubleshoot errors from any of the different APIs

- Record the response body when the response is an error response. All HTTP status codes equal to and greater than 400 are considered error responses.
- Reach out to your account team with the above information for assistance.

### 4.1. Support – FactSet’s Issue Tracker

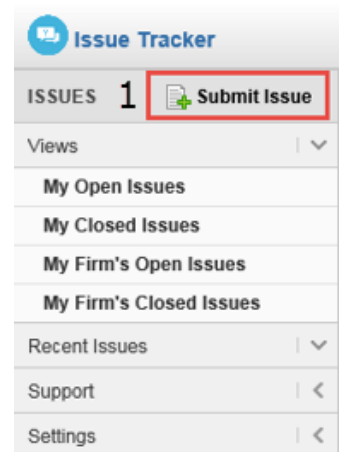
FactSet offers support for Federations and Provisioning via direct submission of support tickets to FactSet Engineers via FactSet’s Issue Tracker at <https://issuetracker.factset.com>.

To gain access to Issue Tracker for Views/White Label and Federation support, contact your FactSet Account Representative.

Engineers and product specialists will handle issues submitted via Issue Tracker. Update notifications will be delivered to your e-mail for any tickets submitted by yourself or colleagues who also have access to Issue Tracker.

### 4.2. Submitting an Issue via Issue Tracker

- Log into <https://issuetracker.factset.com> and click on “Submit Issue”
- From the “Product:” dropdown box, select the Product and then Category for the issue at hand
- Fill in the “Subject:” field with an appropriate title
- You may add to the “Cc:” field any of your colleagues who also have Issue Tracker access
- Fill in the “Description” field with the issue at hand. You may also attach screenshots or other documents to the Issue
- When completed, click on “Send”



Open issues filed by yourself will be available under **“My Open Issues”**; your closed issues will be under **“My Closed Issues”**

Issues opened and closed by others at your firm with Issue tracker will be located under “**My Firm’s Open Issues**” and “**My Firm’s Closed Issues**” respectively.

**Create Issue**

My Products:  All Products:

Product : 2

Subject : 3

Cc : 4

Description 5

6

### 4.3. Support Products

The below Issue Tracker products are available which apply to this documentation. Not all products may be available to you if they are not within the scope of your implementation.

#### Product – Vermilion Support

Categories

**DEV/UAT/PROD/Other environment:** Select the environment where the issue is present

**Requested Priority:** Low/Medium/High/Blocking

### 5. Version Upgrade

FactSet will support old API versions for a limited time. The actual support time will depend on the API and the release stage of the core VRS product. SCIM is available from VRS 21.9+. All breaking changes, functionality additions, and bug fixes across earlier versions will be documented in the Release Notes. FactSet’s API engineering team will work with the clients to ensure smooth transition to newer versions.