### **API Documentation**

## **Get user roles and permission**

Retrieves the roles and permissions for the authenticated user across all teams they belong to.

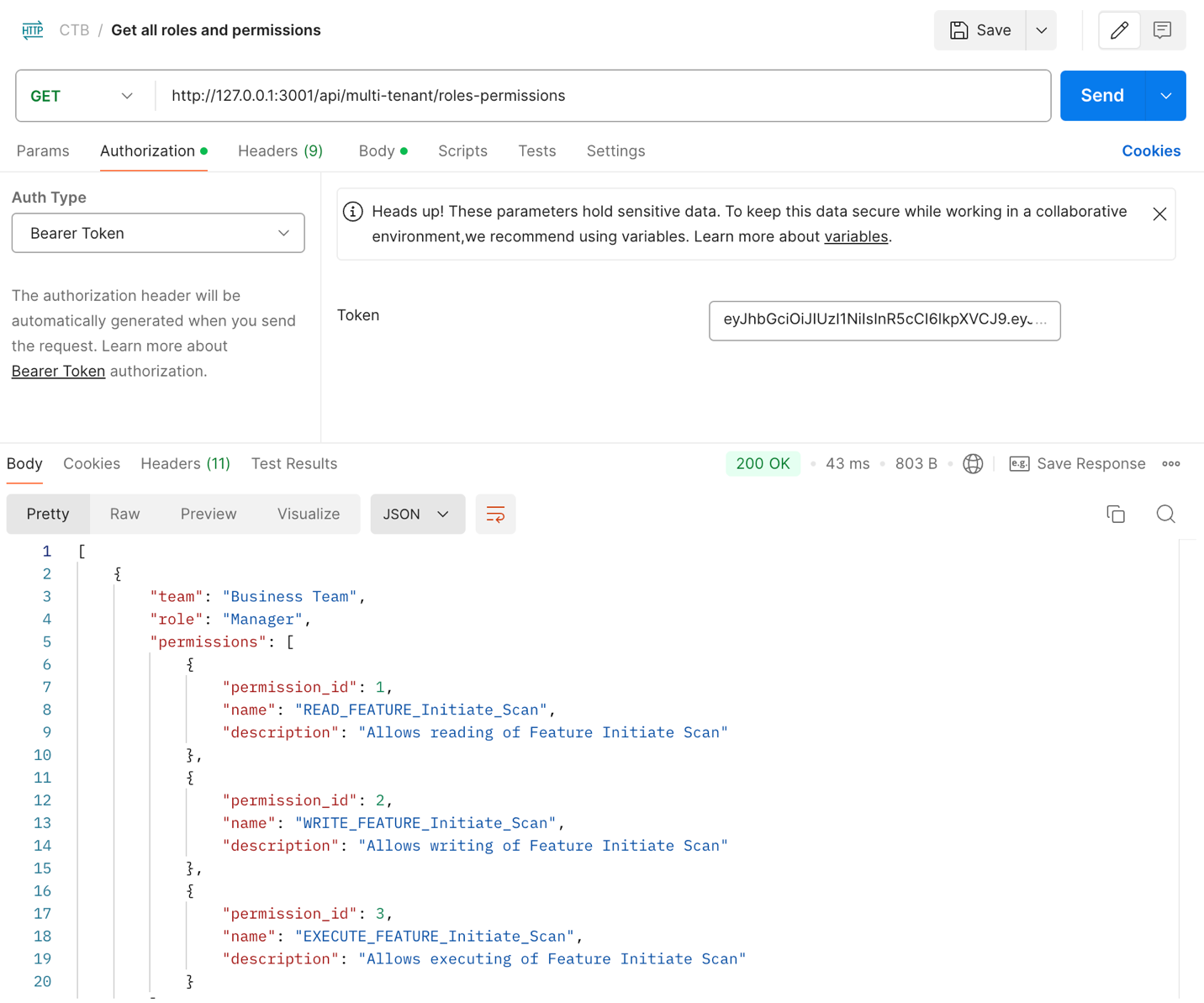
**Endpoint**

GET /api/multi-tenant/roles-permissions

**Authentication**

This endpoint requires authentication. The user's id is extracted from the authentication token.

**Success Response: 200 OK**

[  
 {  
 "team": "Business Team",  
 "role": "Manager",  
 "permissions": [  
 {  
 "permission\_id": 1,  
 "name": "READ\_FEATURE\_Initiate\_Scan",  
 "description": "Allows reading of Feature Initiate Scan"  
 },  
 {  
 "permission\_id": 2,  
 "name": "WRITE\_FEATURE\_Initiate\_Scan",  
 "description": "Allows writing of Feature Initiate Scan"  
 },  
 {  
 "permission\_id": 3,  
 "name": "EXECUTE\_FEATURE\_Initiate\_Scan",  
 "description": "Allows executing of Feature Initiate Scan"  
 }  
 ]  
 }  
]  
 

**Error Response: 500 Internal Server Error**

{

"error": "Something went wrong while fetching the user roles and permissions!"

}

## **Create Team**

**Endpoint**

POST /api/multi-tenant/create-team

Authentication

This endpoint requires authentication. The user's role is used to determine which predefined roles are assigned to the new team.

**Request Body**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type | Required | Description |
| teamName | string | Yes | The name of the new team |
| teamDescription | string | Yes | Description of the team |

**Example Value**

{

"teamName": "Test Researcher Team",

"teamDescription": "This is a test."

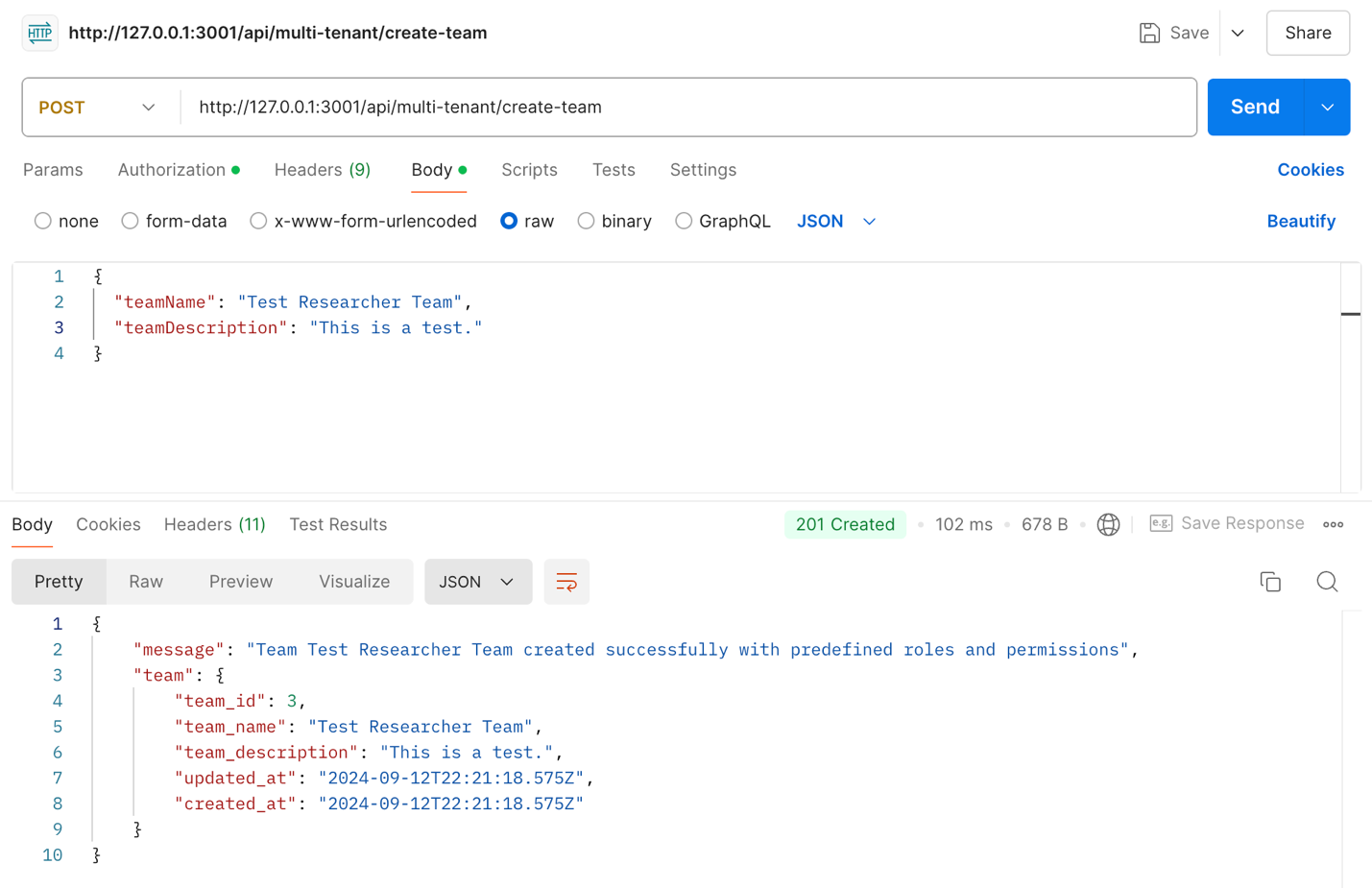
}

**Responses**

**Success Response: 201 Created**

{  
 "message": "Team Test Researcher Team created successfully with predefined roles and permissions",  
 "team": {  
 "team\_id": 3,  
 "team\_name": "Test Researcher Team",  
 "team\_description": "This is a test.",  
 "updated\_at": "2024-09-12T22:21:18.575Z",  
 "created\_at": "2024-09-12T22:21:18.575Z"  
 }  
}

**Error Response: 500 Internal Server Error**  
{  
 "error": "Something went wrong while creating the team. "

}  


## **Edit Team**

**Endpoint**

PUT /api/multi-tenant/edit-team/:id

**Parameters**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **In** | **Description** |
| id | string | path | Unique identifier of the team |

**Request Body**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type | Required | Description |
| teamName | string | Yes | The new name of the team |
| teamDescription | string | Yes | The new description of the team |

**Example Value**

{

"teamName": "Test Researchers",

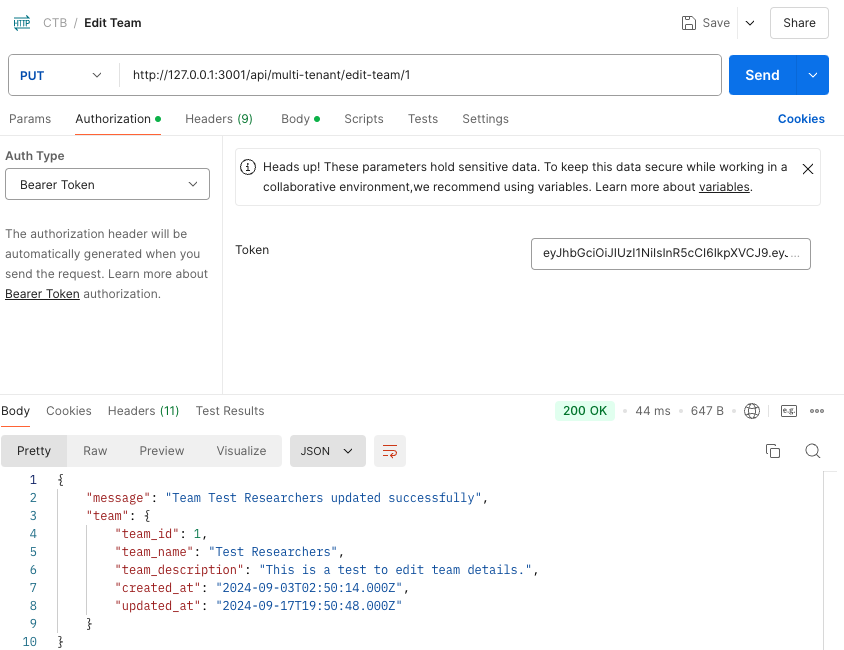
"teamDescription": " This is a test to edit team details. "

}

**Responses**

**Success Response: 200 Success**

{  
 "message": "Team Test Researchers updated successfully",  
 "team": {  
 "team\_id": 1,  
 "team\_name": "Test Researchers",  
 "team\_description": "This is a test to edit team details.",  
 "created\_at": "2024-09-03T02:50:14.000Z",  
 "updated\_at": "2024-09-17T19:50:48.504Z"  
 }  
}



## **Create Team Role**

**Endpoint**

POST /api/multi-tenant/ create-role

Authentication

This endpoint requires authentication. The user's role is used to determine which team-role is created.

**Request Body**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type | Required | Description |
| teamID | int | Yes | The team id of the user |

**Example Value**

{

"teamID": 2

}

**Responses**

**Success Response: 201 Created**

{

"success": **true**,

"data": [

{

"team\_role\_id": 44,

"team\_role\_name": "Admin",

"description": "Team role based on predefined role: Admin",

"team\_id": 2,

"createdAt": "2024-09-19T16:21:56.646Z",

"updatedAt": "2024-09-19T16:21:56.646Z"

},

{

"team\_role\_id": 45,

"team\_role\_name": "Analyst",

"description": "Team role based on predefined role: Analyst",

"team\_id": 2,

"createdAt": "2024-09-19T16:21:56.646Z",

"updatedAt": "2024-09-19T16:21:56.646Z"

},

{

"team\_role\_id": 46,

"team\_role\_name": "Cyber Security Manager",

"description": "Team role based on predefined role: Cyber Security Manager",

"team\_id": 2,

"createdAt": "2024-09-19T16:21:56.646Z",

"updatedAt": "2024-09-19T16:21:56.646Z"

},

{

"team\_role\_id": 47,

"team\_role\_name": "Executive",

"description": "Team role based on predefined role: Executive",

"team\_id": 2,

"createdAt": "2024-09-19T16:21:56.646Z",

"updatedAt": "2024-09-19T16:21:56.646Z"

},

{

"team\_role\_id": 48,

"team\_role\_name": "Intermediate Pentester (Level 2 Security Analyst)",

"description": "Team role based on predefined role: Intermediate Pentester (Level 2 Security Analyst)",

"team\_id": 2,

"createdAt": "2024-09-19T16:21:56.646Z",

"updatedAt": "2024-09-19T16:21:56.646Z"

},

{

"team\_role\_id": 49,

"team\_role\_name": "Junior Pentester (Level 1 Security Analyst)",

"description": "Team role based on predefined role: Junior Pentester (Level 1 Security Analyst)",

"team\_id": 2,

"createdAt": "2024-09-19T16:21:56.646Z",

"updatedAt": "2024-09-19T16:21:56.646Z"

},

{

"team\_role\_id": 50,

"team\_role\_name": "Lead Pentester (Level 3 Security Analyst)",

"description": "Team role based on predefined role: Lead Pentester (Level 3 Security Analyst)",

"team\_id": 2,

"createdAt": "2024-09-19T16:21:56.646Z",

"updatedAt": "2024-09-19T16:21:56.646Z"

},

{

"team\_role\_id": 51,

"team\_role\_name": "Manager",

"description": "Team role based on predefined role: Manager",

"team\_id": 2,

"createdAt": "2024-09-19T16:21:56.646Z",

"updatedAt": "2024-09-19T16:21:56.646Z"

},

{

"team\_role\_id": 52,

"team\_role\_name": "Quality Assurance",

"description": "Team role based on predefined role: Quality Assurance",

"team\_id": 2,

"createdAt": "2024-09-19T16:21:56.646Z",

"updatedAt": "2024-09-19T16:21:56.646Z"

},

{

"team\_role\_id": 53,

"team\_role\_name": "Super Admin",

"description": "Team role based on predefined role: Super Admin",

"team\_id": 2,

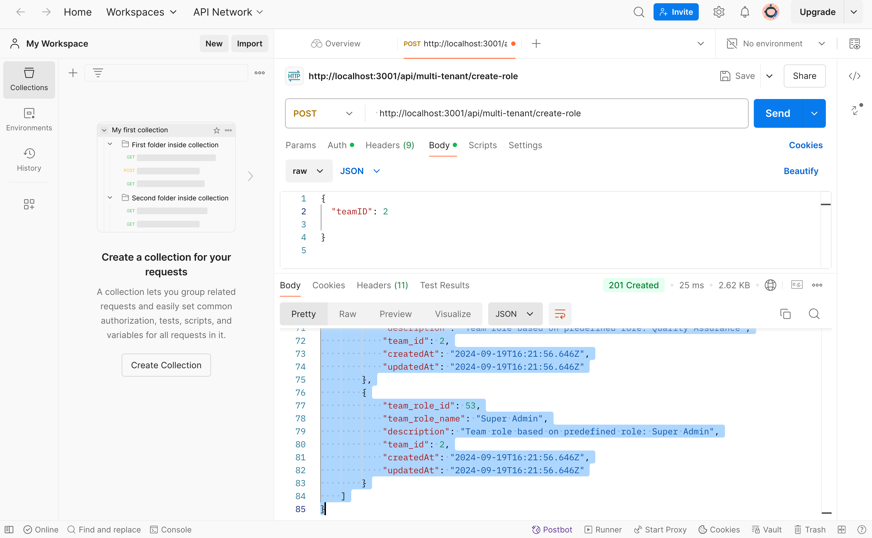
"createdAt": "2024-09-19T16:21:56.646Z",

"updatedAt": "2024-09-19T16:21:56.646Z"

}

]

}



**Error Response: 500 Internal Server Error**

{

"success": **false**,

"message": "Error creating team roles."

}

### **Add permission**

**Endpoint**

POST: /api/v2/permission/add

**Request Format**

{

"name": "EDIT\_REPORT",

"description": "Allows the user to edit reports."

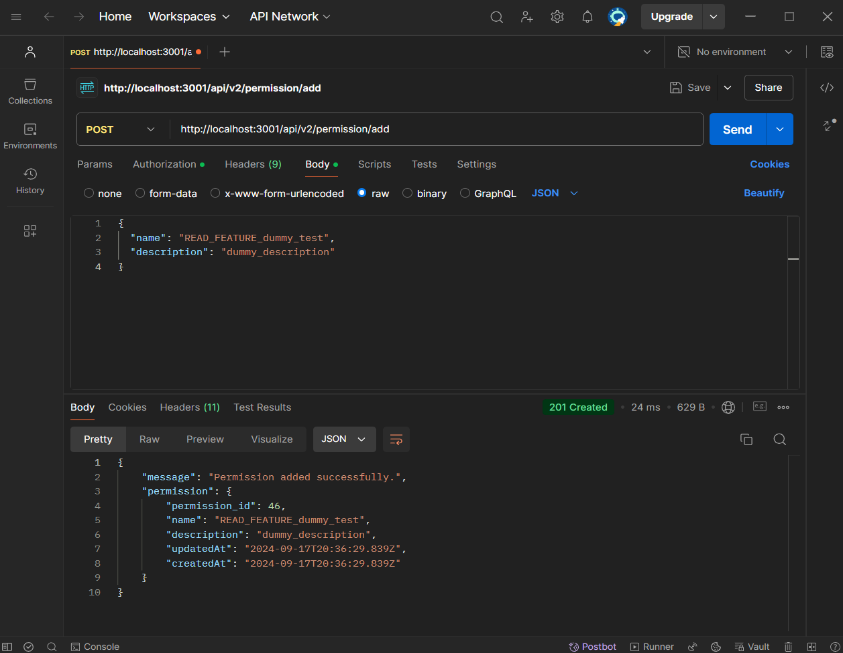
}

**Responses**

Success (201 Created):

[  
 {  
 "message": "Permission added successfully.”,  
 "permissions": [  
 {  
 "permission\_id": 47,  
 "name": "READ\_FEATURE\_Initiate\_Scan",  
 "description": "Allows reading of Feature Initiate Scan",

"createdAt": "2024-09-13T12:00:00Z",

 "updatedAt": "2024-09-13T12:00:00Z"  
 },  
 ]  
 }  
 ]

Add permission testing

### **Delete permission**

**Endpoint**

DELETE: /api/v2/permission/delete/:id

**Request Format**

* **Headers**:
  + Authorization: Bearer <token>
* **Path Parameters**:
  + id (number): The ID of the permission to delete.

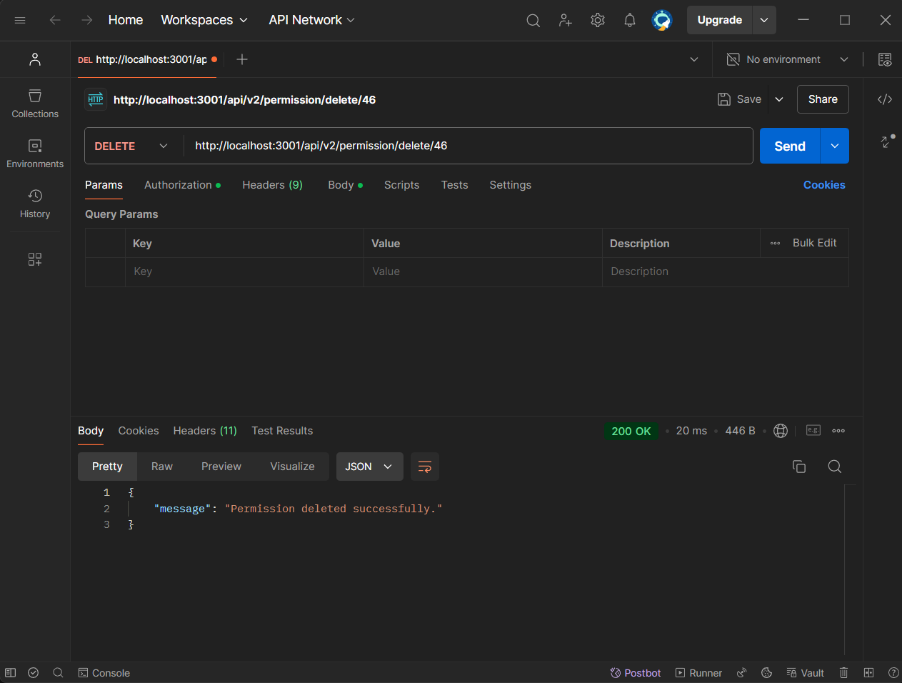
**Responses**

Success (200 OK):

{

"message": "Permission deleted successfully."

}



Delete permission testing

### **Assign permission to the role**

**Endpoint**

POST: /api/v2/permission/assign

**Request Format**

{

"team\_role\_id": 5,

"permission\_id": 47

}

**Responses**

Success (201 Created):

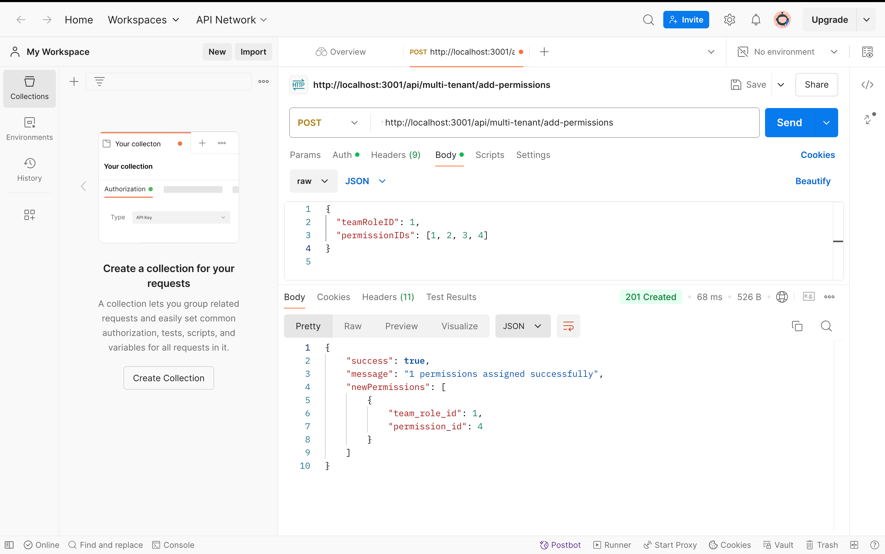
[  
 {  
 "message": "Permission assigned to team role successfully.",  
 "assignment": [  
 {  
 "team\_role\_permission\_id": 12,

"team\_role\_id": 5,

"permission\_id": 47,

"createdAt": "2024-09-13T12:10:00Z",

"updatedAt": "2024-09-13T12:10:00Z"  
 },  
 ]  
 }  
]



### **Remove permission from the role**

**Endpoint**

DELETE/api/multi-tenant/remove-permissions

Authentication

This endpoint requires authentication. The user's role is used to determine what permissions could be removed.

**Request Body**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type | Required | Description |
| teamRoleID | int | Yes | The team role id of this user |
| permisssionIDs | int | Yes | The permissions to be removed |

**Example Value**

{

"teamRoleID": 1,

"permissionIDs": [1, 2, 3, 4]

}

**Responses**

**Success Response: 200 OK**

{

"success": **true**,

"message": "4 permissions removed successfully"

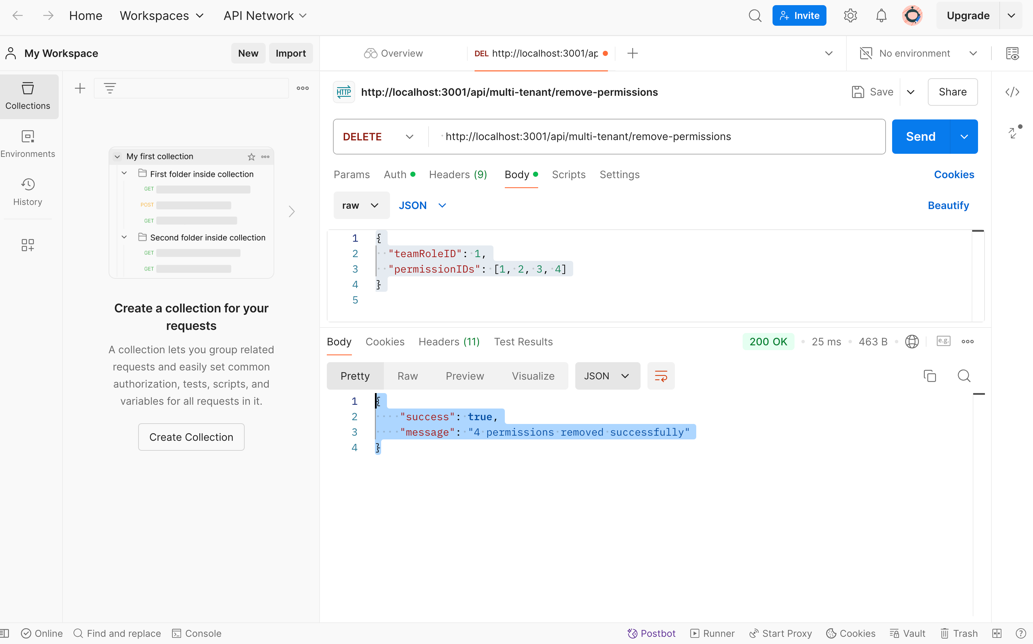
}

**Error Response: 500 Internal Server Error**  
{

"success": **false**,

"message": "Internal server error"

}



### **Invitation Email Services**

**Endpoint**

Send Invitation:

POST /api/invitations/send

Authentication

This endpoint requires authentication. The user's role is used to determine which team role id could be used for invitation.

**Request Body**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type | Required | Description |
| email | string | Yes | The email address for the invited user |
| teamRoleID | int | Yes | The team role id of this user |

**Example Value**

{

"email": "true-ly@hotmail.com",

"team\_role\_id": 82

}

**Responses**

**Success Response: 200 OK**

{

"message": "Invitation sent successfully."

}

**Error Response: 500 Internal Server Error**  
{

"message": "Internal server error"

}

A screenshot of a computer

Description automatically generated

**Endpoint**

Accept Invitation:

POST /api/invitations/accept

**Request Body**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type | Required | Description |
| token | string | Yes | The token received in the invitation |
| username | string | Yes | Username input by user |
| password | string | Yes | Password input by user |

**Example Value**

{

"token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyX2lkIjozLCJpYXQiOjE3MjczMzQ3OTYsImV4cCI6MTcyNzM2MzU5Nn0.uTVtvs28IH51xmgvj5EjN3mcuLfa72zMJRCVsJNtTps",

"username": "testuserforinvite",

"password": "12345678aA."

}

**Responses**

**Success Response: 200 OK**

{

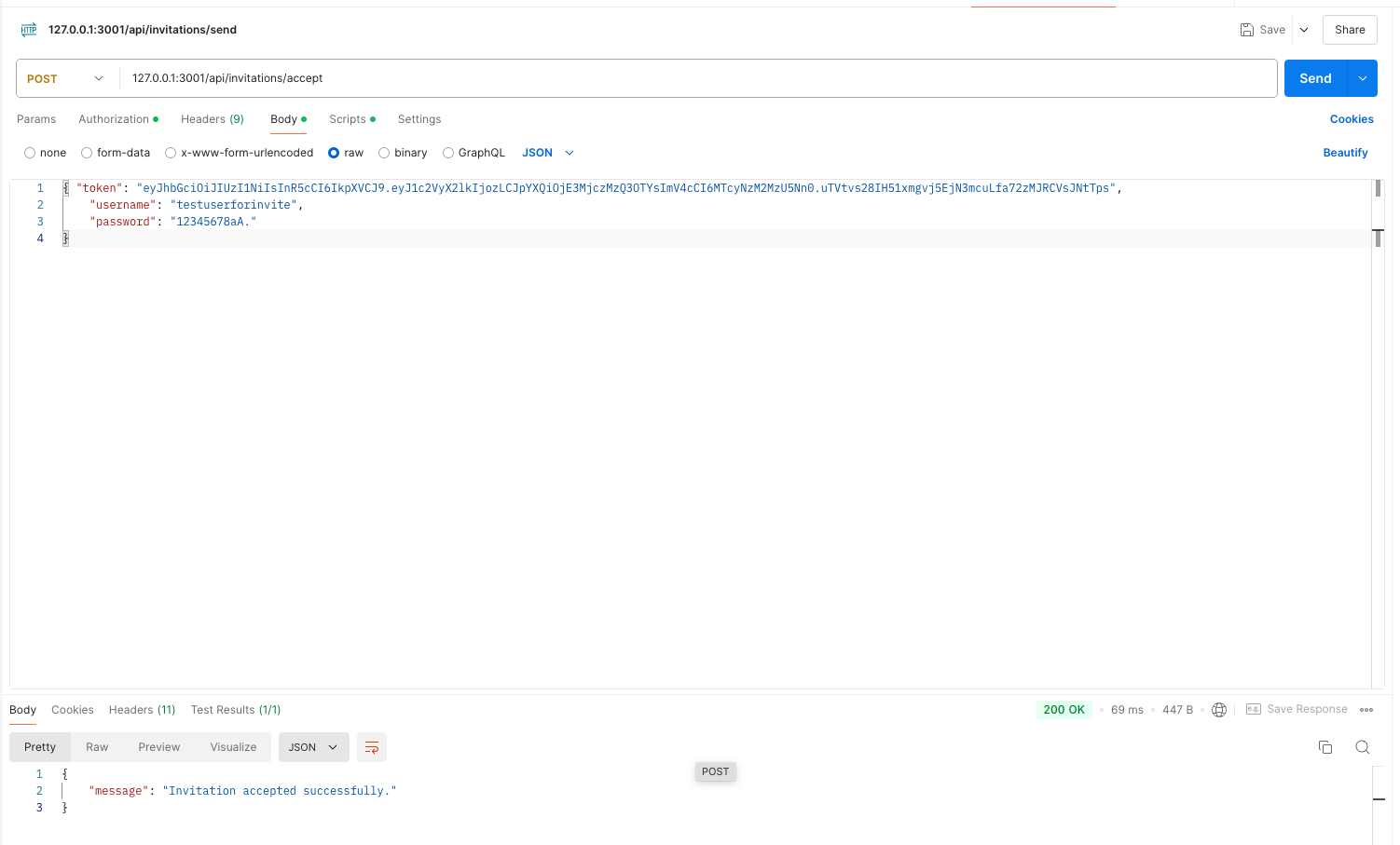
"message": "Invitation accepted successfully."

}

**Error Response: 500 Internal Server Error**  
{

"message": "Internal server error"

}



Resend Invitation:

POST /api/invitations/resend

Authentication

This endpoint requires authentication. The user's role is used to determine which team role id could be used for invitation.

**Request Body**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type | Required | Description |
| email | string | Yes | The email address to resend the invitation |
| teamRoleID | int | Yes | The team role id of this user |

**Example Value**

{

"email": "true-ly@hotmail.com",

"team\_role\_id": 75

}

**Responses**

**Success Response: 200 OK**

{

"message": "Invitation resent successfully."

}

**Error Response: 404 Internal Server Error**  
{

"message": "No pending invitation found for the given email and role."

}

**Error Response: 500 Internal Server Error**  
{

"message": "Internal server error"

}

