# ScimAuthGuard

September 4, 2026

## ScimAuthGuard

**Kind:** Service

**Source:** `atloria-monorepo/apps/api/src/scim/scim-auth.guard.ts`

SCIM bearer-token authentication.

SECURITY — org identity comes ONLY from the token. We hash the presented bearer (sha256), look up a matching, non-revoked ScimToken, and stamp the request with `scimOrgId` from THAT ROW. The SCIM controllers never read an org id from the body or a param — so a token for org A can never touch org B, regardless of what the IdP puts in the payload.

SCIM is a BUSINESS feature: a token whose org is not on BUSINESS is rejected (so a downgrade also stops provisioning).

All failures return an RFC 7644 SCIM error body (via ScimExceptionFilter).

`ScimAuthGuard` authenticates SCIM requests using bearer tokens and establishes the organization identity exclusively from the matched token record. It SHA-256 hashes the presented token, rejects revoked or invalid tokens, and sets `request.scimOrgId` from the token's organization to prevent cross-organization provisioning. Access is limited to organizations on the BUSINESS plan, and failures are returned as RFC 7644 SCIM error responses.

### Methods

| Method | Signature | Returns |
| --- | --- | --- |
| `canActivate` | `canActivate(context: ExecutionContext)` | `Promise<boolean>` |

### Dependencies

- `PrismaService`

- `PlanService`

## Where it refuses work

- `ScimAuthGuard` stops the work with `HttpException` when `!match`.
- `ScimAuthGuard` stops the work with `HttpException` when `!token || token.revokedAt`.
- `ScimAuthGuard` stops the work with `HttpException` when `!meets`.

## Diagram

```
sequenceDiagram
    participant IdP as Identity Provider
    participant Guard as ScimAuthGuard
    participant DB as ScimToken Store
    participant Controller as SCIM Controller

    IdP->>Guard: Request with Authorization: Bearer <token>
    Guard->>Guard: Extract and SHA-256 hash bearer token
    Guard->>DB: Find matching non-revoked ScimToken
    DB-->>Guard: Token record with orgId and subscription status

    alt Token missing, invalid, or revoked
        Guard-->>IdP: RFC 7644 SCIM error response
    else Organization is not on BUSINESS plan
        Guard-->>IdP: RFC 7644 SCIM error response
    else Valid BUSINESS token
        Guard->>Guard: Set request.scimOrgId = token.orgId
        Guard->>Controller: Allow request
        Controller->>Controller: Use request.scimOrgId for all org-scoped work
    end
```

## Usage

```
import { Controller, Get, Req, UseGuards } from '@nestjs/common';
import { Request } from 'express';

import { ScimAuthGuard } from './scim-auth.guard';

@Controller('scim/v2')
@UseGuards(ScimAuthGuard)
export class ScimUsersController {
  @Get('Users')
  async listUsers(@Req() request: Request) {
    // This value is set only by ScimAuthGuard after validating the bearer token.
    const orgId = request.scimOrgId;
```

```
    return {
      Resources: await this.usersService.findScimUsers(orgId),
      totalResults: 0,
    };
  }
}
```

## AI Coding Instructions

- Always derive SCIM organization scope from `request.scimOrgId`; never accept an organization ID from request params, query strings, or payloads.
- Preserve the bearer-token hashing flow: compare the SHA-256 hash of the presented token rather than storing or querying raw tokens.
- Ensure revoked tokens and organizations without the BUSINESS plan are rejected before any SCIM controller logic executes.
- Keep SCIM authentication failures compatible with RFC 7644 by allowing errors to flow through `ScimExceptionFilter`.
- Apply `ScimAuthGuard` to every SCIM endpoint so organization isolation is enforced consistently.

## Relationships

- DEPENDS_ON → `PrismaService`
- DEPENDS_ON → `PlanService`

## Referenced By

- `ScimModule` (MODULE_PROVIDES)