

Docs » Base for Applications Category » Auth Project » Route Protection

Route Protection

Route protection capabilities in Derafu Auth

Anonymous · 09/09/2026

Route Protection

The Derafu Auth package provides route protection through a simple array-based configuration system.

How It Works

The `RouteValidator` class determines which routes require authentication by checking if the requested path starts with any of the configured protected routes.

```
public function requiresAuth(string $path): bool
{
    foreach ($this->config->getProtectedRoutes() as $route) {
        if (str_starts_with($path, $route)) {
            return true;
        }
    }

    return false;
}
```

Configuration

Environment Variable

```
KEYCLOAK_PROTECTED_ROUTES='["/dashboard", "/profile", "/admin"]'
```

Default Protected Routes

If not specified, the package uses these default protected routes:

```
["/dashboard", "/profile", "/admin"]
```

Route Protection Examples

Basic Protection

Protect specific routes:

```
KEYCLOAK_PROTECTED_ROUTES=["/dashboard", "/profile", "/admin"]
```

This configuration will protect:

- `/dashboard` - Exact match.
- `/dashboard/settings` - Starts with `/dashboard`.
- `/profile` - Exact match.
- `/profile/edit` - Starts with `/profile`.
- `/admin` - Exact match.
- `/admin/users` - Starts with `/admin`.

API Protection

Protect API routes:

```
KEYCLOAK_PROTECTED_ROUTES=["/api/v1", "/api/v2"]
```

This will protect all routes starting with `/api/v1` or `/api/v2`:

- `/api/v1/users`
- `/api/v1/users/123`
- `/api/v2/admin`
- `/api/v2/admin/settings`

Admin Panel Protection

Protect admin panel:

```
KEYCLOAK_PROTECTED_ROUTES=["/admin", "/moderator"]
```

This protects:

- `/admin` - Admin panel
- `/admin/users` - User management
- `/admin/settings` - Settings
- `/moderator` - Moderator panel
- `/moderator/comments` - Comment moderation

Limitations

The current implementation has the following limitations:

1. **Prefix Matching Only:** Routes are protected using `str_starts_with()`, so `/admin` will also protect `/admin-panel` and `/administer`.
2. **No Regex Support:** The package does not support regular expressions for route matching.
3. **No Wildcard Support:** There is no support for wildcard patterns like `/admin/*`.
4. **No Negative Patterns:** Cannot exclude specific routes from protection.
5. **No Role-Based Protection:** All protected routes require the same level of authentication. But you can use authorization to protect specific routes based on the user's roles.

Special Routes

The package automatically handles these special routes:

- **Callback Route:** Default is `/auth/callback`.
- **Logout Route:** Default is `/auth/logout`.

These routes are automatically excluded from authentication requirements.

Custom Route Validator

You can create a custom route validator by implementing the `RouteValidatorInterface`:

```
use Derafu\Auth\Contract\RouteValidatorInterface;

class CustomRouteValidator implements RouteValidatorInterface
{
    public function requiresAuth(string $path): bool
    {
        // Your custom logic here
        return str_starts_with($path, '/protected');
    }

    public function isCallbackPath(string $path): bool
    {

```

```
        return $path === '/auth/callback';
    }

    public function isLogoutPath(string $path): bool
    {
        return $path === '/auth/logout';
    }

    public function getProtectedRoutes(): array
    {
        return ['/protected'];
    }

    public function getCallbackRoute(): string
    {
        return '/auth/callback';
    }

    public function getLogoutRoute(): string
    {
        return '/auth/logout';
    }
}
```

Then configure it in your services:

```
services:
    Derafu\Auth\Contract\RouteValidatorInterface:
        class: App\Auth\CustomRouteValidator
```

Last updated on 09/09/2026