Service Contract document

AT-AT Service Contracts

The API Threat Assessment Tool (AT-AT) uses a RESTful architecture and communicates using JSON over HTTP(S). The API is authenticated via Bearer JWT tokens.

`apis.create`

```
{
  "command": "apis.create",
  "data": {
    "name": "string",
    "description": "string",
    "file": "OpenAPI spec file (encoded)"
  }
}
```

Response

- `200 OK`: `{ "api_id": "string" }`
- `400 Bad Request`: Invalid format

Description: Creates a new API entry from an uploaded specification.

`apis.delete`

```
{
  "command": "apis.delete",
  "data": {
    "api_id": "string"
  }
}
```

Response

- `200 OK`: API deleted
- `404 Not Found`: API does not exist

Description: Removes an API and its associated data.

---

apis.details

```
GET /api/apis/:id
```

Response

- `200 OK`: API details and endpoint list
- `404 Not Found`

---

apis.get_all

```
{
  "command": "apis.get_all",
  "data": {
    "user_id": "string"
  }
}
```

Response

- `200 OK`: `[ { "api_id": "string", "name": "string" }, ... ]`
- `404 Not Found`: No APIs found

Description: Retrieves all APIs linked to the given user.

---

apis.import_file

```
POST /api/apis/import/file
FormData:
  - file: OpenAPI file
```

Response

- `200 OK: { api_id }`
- `400 Bad Request`: missing file
- `500 Internal Server Error`

Description: Parses an uploaded OpenAPI/Swagger spec and adds it to the system.

`apis.import_url`

```
{
  "url": "https://example.com/spec.json"
}
```

Response

- `200 OK: { api_id }`
- `400 Bad Request`: invalid/missing URL

`apis.key.set`

```
{
  "command": "apis.key.set",
  "data": {
    "api_key": "string"
  }
}
```

Response

- `200 OK`
- `400 Bad Request`

Description: Stores an API key for later scan use.

`apis.key.validate`

```
{
  "command": "apis.key.validate",
  "data": {
    "api_key": "string"
  }
}
```

Response

- `200 OK`: valid
- `400 Bad Request`: invalid

Description: Validates the format and existence of an API key.

Endpoint Management (Extended)

`apis.update`

```
{
  "name": "string",
  "description": "string"
}
```

`auth.google`

```
{
  "token": "Google OAuth token"
}
```

Response

- `200 OK`: `{ user: {...}, token: "jwt" }`
- `400 Bad Request`: token missing
- `500 Internal Server Error`

Description: Handles Google login and user creation if needed.

`auth.login`

```
{
  "username": "string", // or
  "email": "string",
  "password": "string"
}
```

Response

- `200 OK`: `{ user: {...}, token: "jwt" }`
- `401 Unauthorized`: invalid credentials
- `500 Internal Server Error`

Description: Authenticates user credentials, returns session token on success.

`auth.logout`

```
{
  "command": "auth.logout",
  "data": {}
}
```

Response

- `200 OK`: Logout successful
- `500 Internal Server Error`

Description: Logs the current user out and invalidates their token.

API Management (Extended)

`connection.test`

```
{
  "command": "connection.test",
  "data": {}
}
```

Response

- `200 OK`: connection confirmed

Description: Verifies the daemon is active and responsive.

`dashboard.alerts`

```
{
  "command": "dashboard.alerts",
  "data": {}
}
```

Response

- `200 OK`: { “alerts”: […] }

Description: Returns dashboard alerts such as scan failures or detected threats.

Scan Services (Extended)

`dashboard.metrics`

```
GET /api/dashboard/metrics
```

Response

- `200 OK`: { metrics }
- `500 Internal Server Error`

Description: Returns scanning trends, usage, and recent activity.

`dashboard.overview`

```
GET /api/dashboard/overview
```

Response

- `200 OK`: { overview_data }
- `500 Internal Server Error`

Description: Returns system-wide summary (total APIs, scans, threats).

`endpoints.details`

```
{
  "command": "endpoints.details",
  "data": {
    "endpoint_id": "string"
  }
}
```

Response

- `200 OK`: endpoint metadata
- `404 Not Found`

Description: Retrieves full metadata for a specific API endpoint.

`endpoints.flags.add`

```
{
  "path": "/api/resource",
  "method": "GET",
  "flags": "BOLA"
}
```

Response

- `200 OK`
- `400 Bad Request`: invalid flag or missing fields

Description: Adds OWASP 10 flags to specific endpoints.

`endpoints.flags.remove`

```
{
  "command": "endpoints.flags.remove",
  "data": {
    "endpoint_id": "string",
    "flags": "BOLA"
  }
}
```

Response

- `200 OK`
- `400 Bad Request`

Description: Removes flags from an endpoint.

Dashboard (Extended)

`endpoints.list`

```
{
  "api_id": "string"
}
```

Returns: all parsed endpoints

---

endpoints.tags.add

```
{
  "command": "endpoints.tags.add",
  "data": {
    "endpoint_id": "string",
    "tags": ["tag1", "tag2"]
  }
}
```

Response

- 200 OK
- 400 Bad Request

Description: Adds tags to the given endpoint.

---

endpoints.tags.remove

```
{
  "command": "endpoints.tags.remove",
  "data": {
    "endpoint_id": "string",
    "tags": ["tag1"]
  }
}
```

Response

- 200 OK
- 400 Bad Request

Description: Removes specific tags from an endpoint.

`endpoints.tags.replace`

```
{
  "path": "/user/:id",
  "method": "DELETE",
  "tags": ["sensitive", "admin-only"]
}
```

Response

- `200 OK`
- `400 Bad Request`: invalid input

`reports.details`

```
{
  "command": "reports.details",
  "data": {
    "report_id": "string"
  }
}
```

Response

- `200 OK`: detailed report content

Description: Retrieves contents of a specific report.

Templates & Tags

`reports.download`

```
GET /api/reports/:id/download?report_type=technical
```

Returns: file (PDF or text)

Error Schema

All endpoints return standardized error responses:

```
{
  "status": "error",
  "message": "Detailed error message"
}
```

Timeout & Limits

- All scan jobs are capped at **60 seconds**.
- Endpoints are rate-limited to prevent abuse.

`reports.list`

```
{
  "command": "reports.list",
  "data": {}
}
```

Response

- `200 OK`: list of reports

Description: Fetches list of previous scan reports.

`scan.create`

```
{
  "client_id": "uuid",
  "scan_profile": "OWASP_API_10"
}
```

Returns: `scan_id`, `results_count`

`scan.list`

```
{
  "command": "scan.list",
  "data": {}
}
```

Response

- `200 OK`: List of scans

Description: Lists all scans performed by the user.

`scan.progress`

```
{
  "command": "scan.progress",
  "data": {
    "scan_id": "string"
  }
}
```

Response

- `200 OK`: progress data
- `404 Not Found`

Description: Shows real-time progress of a scan.

---

scan.results

```
GET /api/scans/:id/results
```

Returns: vulnerability data + endpoint details

---

scan.start

```
{
  "command": "scan.start",
  "data": {
    "api_name": "string",
    "scan_profile": "string"
  }
}
```

Response

- `200 OK`: scan started
- `503`: scan in progress

Description: Resumes or starts a new scan session.

---

scan.stop

```
{
  "command": "scan.stop",
  "data": {
    "scan_id": "string"
  }
}
```

Response

- `200 OK`: stopped
- `404`: not found

Description: Stops an active scan.

Reports (Extended)

`tags.list`

```
{
  "command": "tags.list",
  "data": {}
}
```

Response

- `200 OK`: available tags

Description: Lists tags used to classify endpoints.

`templates.details`

```
{
  "command": "templates.details",
  "data": {
    "template_id": "string"
  }
}
```

Response

- `200 OK`: template information

Description: Fetches full detail of a template.

`templates.list`

```
{
  "command": "templates.list",
  "data": {}
}
```

Response

- `200 OK`: list of templates

Description: Shows available scan profiles/templates.

`templates.use`

```
{
  "command": "templates.use",
  "data": {
    "template_id": "string",
    "api_id": "string"
  }
}
```

Response

- `200 OK`: scan started

Description: Launches a scan based on template rules.

User Management

`user.profile.update`

```
{
  "command": "user.profile.update",
  "data": {
    "username": "string",
    "email": "string"
  }
}
```

Response

- `200 OK`

Description: Updates user profile.

`user.settings.get`

```
{
  "command": "user.settings.get",
  "data": {}
}
```

Response

- `200 OK`: settings

Description: Fetches user's current settings.

`user.settings.update`

```
{
  "command": "user.settings.update",
  "data": {
    "notifications": true
  }
}
```

Response

- `200 OK`

Description: Updates user's notification preferences.

Daemon Connectivity

Error Schema

All endpoints return standardized error responses:

```
{
  "status": "error",
  "message": "Detailed error message"
}
```

Timeout & Limits

- All scan jobs are capped at 60 seconds.
- Endpoints are rate-limited to prevent abuse.