AiVRIC API Authentication & Rate Limits

Version 1.0
Last Updated: December 2025

1.0 Overview

The AiVRIC API provides programmatic access to findings, posture data, configurations, and automation workflows within your AiVRIC tenant. All API access requires authenticated requests using one of the supported mechanisms outlined below.

This document describes:

- Supported authentication methods
- Token lifecycle and security considerations
- Request signing and header requirements
- Rate limits & throttling behavior
- Best practices for large-scale integrations
- Error codes and troubleshooting patterns

AiVRIC does **not** expose an open-source API or CLI. All API access occurs through authenticated, tenant-scoped channels managed within the AiVRIC platform.

2.0 Authentication Methods

AiVRIC supports two authentication models:

2.1 OAuth2 Client Credentials

Recommended for:

- SIEM ingestion
- Scheduled ETL pipelines
- Automation and service accounts

AiVRIC issues a client ID and client secret within the **Admin → API Access** panel. These credentials must be exchanged for a short-lived bearer token.

Token Endpoint

```
POST https://api.aivric.app/oauth2/token
```

Request

```
grant_type=client_credentials
client_id=YOUR_CLIENT_ID
client_secret=YOUR_CLIENT_SECRET
```

Response

```
{
  "access_token": "eyJhbGciOi...",
  "token_type": "Bearer",
  "expires_in": 3600
}
```

2.2 Personal Access Tokens (PAT)

Recommended for:

- Ad-hoc queries
- Local tooling development
- Low-frequency admin operations

PATs are tied to the user who created them and follow that user's RBAC entitlements.

Header Usage

```
Authorization: Bearer <PAT>
```

PATs do **not** refresh automatically and must be regenerated if revoked or expired.

3.0 Request Requirements

3.1 Authentication Header

All API calls require the following header:

```
Authorization: Bearer <token>
```

3.2 Accept & Content-Type

JSON is the only supported format.

```
Accept: application/json
Content-Type: application/json
```

3.3 Idempotency Key (Optional)

For POST/PUT calls that modify configuration, use:

```
Idempotency-Key: <uuid>
```

This prevents duplicates during retries or pipeline failures.

4.0 Token Lifecycle & Security

4.1 Token Expiration

- OAuth tokens default to **1 hour (3600 seconds)**
- PATs follow user-defined expiry (7–365 days)

4.2 Token Revocation

Tokens are automatically revoked when:

- The associated user is disabled
- RBAC roles are modified
- An administrator triggers a manual revoke
- AiVRIC detects anomalous access patterns

4.3 Recommended Storage

- Never hardcode client secrets in CI pipelines
- Use a secrets manager (Azure Key Vault, AWS Secrets Manager, GCP Secret Manager)

4.4 Recommended Renewal Pattern

Your automation should refresh tokens **5 minutes before** expiration.

Example pseudo-code:

```
if token_expiry - now < 300:
  refresh_token()
```

5.0 Rate Limits

AiVRIC enforces rate limits globally per tenant. Rate limits apply to **all tokens**, including PATs, OAuth tokens, and internal service accounts.

5.1 Default Rate Limits

API Category	Limit	Window
Findings API	600 requests	per minute
Posture Snapshots	60 requests	per minute
Configuration APIs	120 requests	per minute
Evidence/Exports	30 requests	per minute
Authentication (Token)	20 requests	per minute

6.0 Throttling Behavior

If a client exceeds rate limits, AiVRIC responds with:

6.1 429 Response

```
HTTP/1.1 429 Too Many Requests
Retry-After: 5
X-RateLimit-Limit: 600
X-RateLimit-Remaining: 0
X-RateLimit-Reset: 1702254231
```

6.2 Retry Strategy

Use **exponential backoff**:

1s → 2s → 4s → 8s → 16s (max)

6.3 Long-Running Workloads

For high-volume ingestion:

- Use pagination
- Avoid fetching full snapshots each request
- Cache responses locally
- Only poll deltas / since-last-run offsets

7.0 Pagination & Sorting

7.1 Cursor-Based Pagination

Example:

GET /findings?limit=500&cursor=abc123

7.2 Sort Filters

sort=created_at:desc
sort=severity:asc

8.0 Service Accounts & RBAC

8.1 Service Account Behavior

Service accounts created under **API Access → Service Accounts** inherit permissions assigned to their role.

8.2 Available Roles

- **Read-Only**
- **Security Analyst**
- **Administrator**

8.3 Fine-Grained Access

Each API category can be toggled per service account:

- Findings
- Posture
- Evidence
- Configuration
- Integrations

9.0 Common Errors

Error	Meaning	Resolution
401 Unauthorized	Token invalid or missing	Refresh token, check header
403 Forbidden	RBAC denies resource	Adjust roles/permissions
429 Too Many Requests	Rate limit exceeded	Backoff + retry
500 Internal Error	AiVRIC processing error	Retry with idempotency key
503 Service Unavailable	Maintenance window	Retry after “Retry-After”

10.0 Best Practices

10.1 For Automation

- Refresh OAuth tokens proactively
- Use idempotency keys
- Cache findings
- Minimize full exports

10.2 For SIEM Integrations

- Push deltas, not full datasets
- Honor the Retry-After header
- Rotate tokens every 90 days

10.3 For High-Security Environments

- Store secrets in a secure vault
- Enforce MFA for all admins

- Rotate PATs frequently
- Audit token usage quarterly

11.0 Example: Query Latest Findings

```
GET https://api.aivric.app/v1/findings?limit=200&severity=high
Authorization: Bearer eyJhbGciOi...
Accept: application/json
```

Response

```
{
  "items": [...],
  "cursor": "next_abc123"
}
```

12.0 Support

For assistance:
support@aivric.com
https://docs.aivric.com