

ALGORITHMIA ENTERPRISE

# **SECURITY & COMPLIANCE**

## **OVERVIEW**

Algorithmia Enterprise is the foundation layer for intelligent software. It turns complex services and machine learning models into REST APIs, centralizes them for ease of discoverability, and monitors them from a single dashboard. Companies use Algorithmia Enterprise to reduce duplication of effort between siloed teams and accelerate go-to-market for AI-driven products.

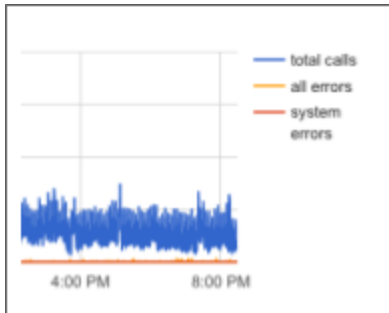
Algorithmia Enterprise is designed from the ground-up to meet the most stringent regulatory and corporate compliance requirements. Our customers' needs span a wide range of complexity, from online retail stores with internet-facing endpoints, to government and finance entities with locked-down classified environments.

This document gives a brief overview on the following topics:

- Cluster Configuration
- Three-Layer Permissions
- Organizations & QA Workflows
- Audit Trails

## Cluster Configuration

Many use cases require code to run inside a private network, behind complex firewall rules, and for data to never leave specific boundaries. Algorithmia Enterprise is designed with a flexible infrastructure that is compatible with government-grade security requirements.



### Public Cloud vs. On-premises

Algorithmia Enterprise is cloud-agnostic and can be deployed on any of the major public clouds (AWS, Google, Azure) as well as on-premises via OpenStack. All deployments will be isolated within their own VPC or combined with an existing VPC.

<b>wkr-565bc6f7</b> avg cpu: 0.03 avg memory: 0.91 used disk: 0.76	10.0.116.251 cloud: aws region: us-east-1 zone: us-east-1d	X
<b>wkr-de610c80</b> avg cpu: 0.33 avg memory: 0.29 used disk: 0.61	10.0.148.64 cloud: aws region: us-west-2 zone: us-west-2c	X
<b>wkr-fa8ea761</b> avg cpu: 0.71 avg memory: 0.52 used disk: 0.25	10.0.26.246 cloud: aws region: us-west-2 zone: us-west-2a	X

### Compute and Data Sovereignty

Algorithmia Enterprise can be configured to work as a multi-region cluster to comply with data sovereignty requirements within regulated industries. A typical setup allows two compute pools, one for each region, and a URL prefix for each region, such as:

<https://<region>.api.acme-codex.internal/api/user/algo>

deeplearning/Inception4/0.1.3 (GPU) user23531, slot3965763	Running
nlp/SentimentAnalysis/0.1.2 (CPU) user76639, slot1938503	Running
nlp/Word2Vec/0.1.1 (CPU) user98752, slot8875442	Running
deeplearning/Inception4/0.1.3 (GPU) user23531, slot5525893	Loaded
deeplearning/Inception4/0.1.3 (GPU) user23531, slot878256	Loaded

### Session Isolation

A job (or API calls) on Algorithmia Enterprise will always operate in its own memory space and will never share or leak memory to other jobs. Each API call instantiates a dedicated Docker container which is destroyed after execution, for perfect isolation at a peta-scale performance.

## Three-Layer Permissions

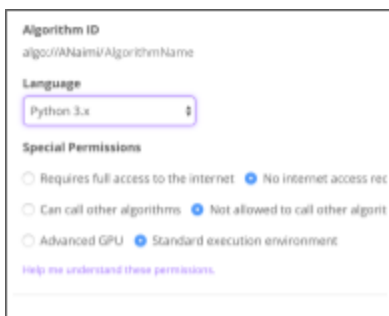
Users want to understand and avoid unnecessary risk to their data when building an integration with another service, even if it was developed by their colleagues. To that end, Algorithmia Enterprise implements permissions on three layers: API keys, algorithms, and data sources.



The screenshot shows a 'New API Key' configuration form. It includes a 'Label' field with the value 'project-one'. The 'Algorithm access' section has a text input containing 'algos/docs/javaAddOne' or 'algos/docs/\*\*' and a dropdown menu set to 'algos/\*\*'. Below this, there are two radio buttons for 'Allow calling algorithms from': 'Native clients (curl, java, etc.)' (selected) and 'Web browser (message CORS)'. The 'Data access' section has a dropdown menu set to 'All data sources'.

### API Key Permissions

Users are expected to create multiple API keys under their profile, one for each project or experiment. Each API key is individually auditable, revocable, limited to access specific algorithms and given explicit read/write permissions over data sources.



The screenshot shows an 'Algorithm ID' configuration form. It includes a text input for 'Algorithm ID' with the value 'algo://Name/AlgorithmName'. The 'Language' dropdown is set to 'Python 3.x'. The 'Special Permissions' section has four radio buttons: 'Requires full access to the internet' (unselected), 'No internet access req' (selected), 'Can call other algorithms' (unselected), and 'Not allowed to call other algorit' (selected). There are also radio buttons for 'Advanced GPU' (unselected) and 'Standard execution environment' (selected). A link 'Help me understand these permissions.' is visible at the bottom.

### Algorithm Permissions

Authors must specify whether their algorithm requires (1) access to network, and (2) access to call other algorithms. Algorithms created without those permissions will be executed in sandboxed Docker containers that do not have access to those resources. Permissions are clearly displayed on the algorithm page and made available to the consumer.



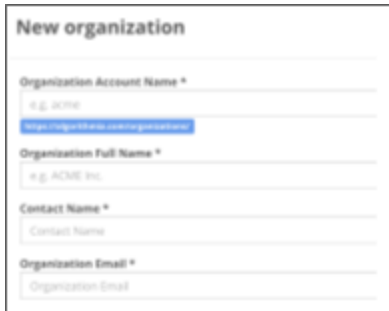
The screenshot shows a 'User/Collection' permissions configuration form. It includes a 'Read Access' dropdown set to 'Private (only me)' and a 'Write Access' dropdown set to 'Public (anybody)'. Below this is a list of '10 Files' with columns for file name and URL. The files listed are: '132-stal-books.txt', 'jfr-demos.txt', 'lancacard', 'math1.jpg', and 'test-test.txt'.

### Data Permissions

Similar to algorithms, a data source (or a data collection) can be configured to allow or disallow Read/Write access from other users. This is especially valuable when using Algorithmia Enterprise within a multi-business line environment where internal data regulations enforce a specific compliance scheme.

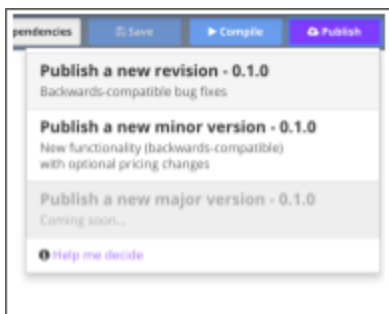
## Organizations & QA Workflows

Large organizations are broken down to teams with different mandates and access rights. Algorithmia Enterprise is designed to adhere to that design while at the same time bring an unprecedented level of collaboration between siloed groups.



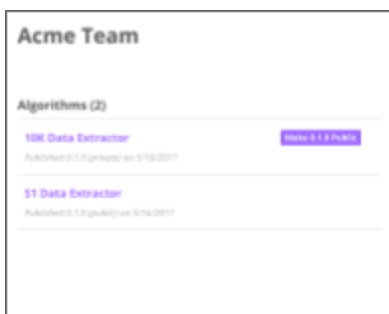
### Teams & Organizations

Users can create or join teams, which gives them the ability to create algorithms under that team's name instead of their own. This allows algorithms to belong to and be maintained by a group of people (department or business line) instead of an individual (i.e. an engineer).



### Private vs. Public Algorithms

Algorithms can be marked as private or public. Private algorithms are only accessible to their owners, which could be an individual or a team. Public algorithms are accessible to all users on the Algorithmia Enterprise platform.

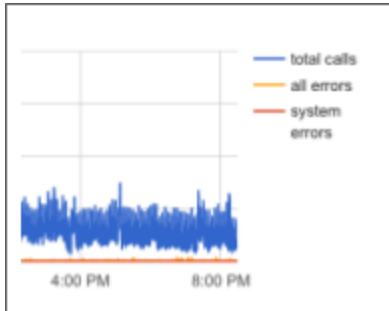


### QA Publishing Workflow

Organizations can choose to enforce a publishing workflow whereby algorithms can only be made accessible to outsiders after it is approved by a compliance officer. This ensures a minimum-level of quality assurance in the public pool of algorithms, while still allowing individuals to experiment privately.

## Audit Trails

Monitoring, tracing, and preventing suspicious activity is critical to all companies, especially in industries with high level of compliance and regulation. Algorithmia Enterprise can be configured to capture and trace any activity.



### API Logging and Verbosity

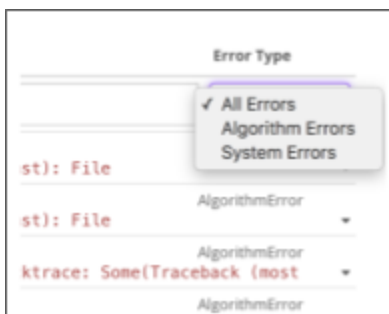
Algorithmia Enterprise logs every API call, showing what user used what API key to call what algorithm and version. Admins can configure the logging module to capture more or less (such as full or partial inputs and outputs) depending on the use case.



Date	Date (UTC)	Algorithm Called
5/15/2017		deepLearning/InceptionV4@1.2
5/15/2017		deepLearning/InceptionV4@1.3
5/15/2017		api/DAT@0
5/15/2017		api/SocialSentimentAnalysis@1.4
5/15/2017		amazon/RemoveOutliers@1.0
5/15/2017		api/DevMock@1.06711310a415cdaa5076d918a43c2441c
5/15/2017		deepLearning/InceptionV4@1.3
5/15/2017		api/DAT@0
5/15/2017		api/SocialSentimentAnalysis@1.4

### Usage Attribution

Algorithmia Enterprise measures how much each user in the platform contributes to the total cluster utilization, allowing administrators to understand how different teams and specific individuals are contributing to total computing costs.



Error Type
<input checked="" type="checkbox"/> All Errors
<input type="checkbox"/> Algorithm Errors
<input type="checkbox"/> System Errors
st): File
AlgorithmError
st): File
AlgorithmError
ktrace: SomeTraceback (most
AlgorithmError

### Error Logs

Every exception anywhere in the platform is captured, including errors from user-generated algorithms. Exceptions can be configured to capture full or partial input, error message, stack trace, and be connected to a notification trigger to take corrective action.