Cyber Knowledge Exchange Platform - User Manual (Complete)

Cyber Project Team

February 21, 2026

Contents

Cyber Knowledge Exchange Platform: User Manual

Version: 1.0.0 (Final - Complete) Platform Version:** 2.1.0 **Last Updated:** February 21, 2026

At its core, Cyber enables:

1. **Persistent Knowledge Exchange** — Organizations create and manage *notebooks* (domain-specific knowledge spaces) that evolve through collaborative contributions from multiple users.
2. **Causal Time Without Clock Synchronization** — Instead of relying on wall-clock timestamps, the platform uses *causal positions* (monotonic sequence numbers per notebook) to establish a consistent order of events. This is essential in distributed, air-gapped, or high-latency environments where synchronized clocks cannot be guaranteed.
3. **Entropy-Based Knowledge Integration** — Every entry carries an *integration cost* (a measure of its “resistance to change”) computed from how well the entry aligns with existing knowledge in the notebook. Over time, entries accumulate “stability” through integration with related content, providing a time arrow without external clocks.
4. **Bell-LaPadula Security Model** — Information is classified at five levels (PUBLIC → TOP_SECRET) and compartmented (restricted to specific clearance categories). The platform enforces strict *information flow control*: classified information never flows to less-classified recipients.
5. **Multiple Interfaces** — Access Cyber through a web-based **Blazor Server UI**, programmatic **MCP integration** (for Claude Desktop AI workflows), or **REST API** for custom integrations.

Why Cyber Exists

Traditional knowledge management systems (wikis, note apps, content management systems) were designed for *open collaboration* in unclassified environments. They assume:

- All users have similar clearance levels
- Information has uniform sensitivity
- Timestamps are reliable global ordering mechanisms
- Changes propagate instantly to all participants

Cyber rejects these assumptions. It's built for environments where:

- **Security compartmentation is non-negotiable** — Healthcare (HIPAA), military (classified), finance (PCI-DSS), research (ITAR) all require strict separation of sensitive information.

- **Global clock synchronization is impractical** — Distributed teams, air-gapped networks, and high-latency links make wall-clock ordering unreliable. Causal ordering is more robust.

- **Knowledge integration matters** — The value of a fact in a knowledge base depends on how well it connects to related facts. Entropy metrics help identify "orphan" entries or contradictions that need human attention.

- **Compliance is mandatory** — Auditors need to see *who* accessed *what when*, with cryptographic proof. Every operation is logged and immutable.

Core Concepts at a Glance

Before diving into workflows, familiarize yourself with these foundational concepts:

1. Notebooks A notebook is a **domain-specific knowledge space** owned by an organization or team. Think of it as a classified database with its own access control list, retention policies, and security boundaries.

Examples: - "Marketing Strategic Initiatives" (PUBLIC classification) - "R&D Cancer Research" (CONFIDENTIAL, Medical Research compartment) - "Operations Security Incidents" (TOP_SECRET, Infrastructure compartment)

Each notebook has: - **Owner group** — The team/department that created and manages it - **Classification level** — Inherited from owner or explicitly set (future) - **Compartments** — Optional security categories that further restrict access - **Retention policy** — How long entries are kept - **Access tiers** — Four levels of permission (existence/read/read+write/admin)

2. Entries An entry is a **unit of knowledge** in a notebook — equivalent to a wiki page, forum post, or document. Entries are:

- **Content-agnostic** — Store any MIME type (text, JSON, markdown, PDF, binary)
- **Immutable** — Once written, entries cannot be deleted or edited in-place. Instead, you *revise* them, creating a new version that supersedes the old.
- **Cryptographically signed** — Every entry includes an Ed25519 signature proving who created it and that it hasn't been tampered with.
- **Causally linked** — Entries reference related entries, building a directed graph of knowledge relationships. Unlike typical wikis, links can be *cyclic*, allowing for feedback loops in knowledge representation.

Entry structure:

```json
{
  "id": "entry_abc123",
  "position": 42,
  "notebook_id": "nb_xyz789",
  "content": "Base64-encoded or raw binary content",
  "content_type": "text/markdown; charset=utf-8",
  "author_id": "author_public_key_hash",
  "signature": "Ed25519 signature bytes",
  "topic": "organization/team/security/access-control",
  "references": ["entry_ref1", "entry_ref2"],
  "created_at": 1708501800,
  "integration_cost": 2.15,
  "status": "probation | integrated | contested"
}
```

3. Causal Positions Instead of relying on timestamps, Cyber uses **causal positions** — monotonically increasing sequence numbers per notebook.

Why? In distributed systems: - Clocks drift, get out of sync, or are deliberately unreliable - Different datacenters/organizations have different time references - "First" and "last" become ambiguous in high-latency networks

Causal positions solve this: Position 42 always comes before Position 43 within a notebook, regardless of when they were actually created or if clocks are skewed.

Practical implication: When you query recent changes, you use causal positions, not timestamps.

4. Integration Cost & Entropy The platform computes an **integration cost** for each entry — a measure of how well it fits with existing knowledge.

How it works:

1. New entry is submitted
2. System compares it (via TF-IDF similarity) against all other entries in the notebook
3. Clusters are formed, measuring *coherence* (how similar related entries are)
4. Integration cost = measure of how much the new entry disrupts existing coherence
   - High cost: Entry is novel, contradicts existing knowledge, or is an outlier
   - Low cost: Entry naturally fits with existing related entries

Why it matters:

- **High-cost entries flag disagreements** — Multiple competing theories get high costs until one achieves dominance
- **Stable entries accumulate low cost** — Over time, well-integrated entries become "anchors" that new entries must align with
- **Retroactive cost propagation** — When a contradictory entry is integrated, previously-high-cost alternatives may increase in cost
- **Time without clocks** — Integration cost provides a "time arrow": entries that are more integrated are "older" (more established) in the community consensus

Entry status values:

Status	Meaning
`probation`	New entry, cost still being calculated, not yet integrated
`integrated`	Stable entry with low cost, part of established knowledge
`contested`	High cost, contradicts other entries, multiple competing theories

5. Security Labels & Classification Every organization and entry has a **security label** consisting of:

1. **Classification level** (Five-level hierarchy):
   - `PUBLIC` — No restrictions (accessible to anyone)
   - `CONFIDENTIAL` — Internal use only
   - `SECRET` — Restricted distribution
   - `TOP_SECRET` — Severe impact if disclosed
   - (Organization-defined custom levels)
2. **Compartments** (Optional security categories):
   - Examples: "Medical Research", "Infrastructure", "Strategic Planning"
   - A user must be explicitly cleared for each compartment they access
   - Information can flow only to users whose clearance dominates the classification + compartments

Bell-LaPadula Dominance Rule:

User clearance `C1` dominates `C2` if: - `C1.level   C2.level` AND - `C2.compartments   C1.compartments`

Example: User cleared for `TOP_SECRET / {Medical, Infrastructure}` dominates: - `SECRET / {Medical}   - TOP_SECRET / {Infrastructure}   - TOP_SECRET / {Medical, Infrastructure}   - TOP_SECRET / {Medical, Infrastructure, Strategic}` (compartment mismatch)

6. Federated Identity Users are identified by **cryptographic public keys** (Ed25519), not usernames:

- **No central PKI** — Organizations manage their own key issuance
- **Portable identity** — Same key works across multiple Cyber instances
- **Cryptographic proof** — Every operation is signed, proving the user's identity without relying on the server

AuthorId = Hash of user's public key. This ensures different keys = different identities, even if they have the same name.

7. Access Tiers Notebooks support four access tiers for each principal (user or group):

Tier	Can Exist	Can Read	Can Write	Can Admin
Existence				
Read				
Read+Write				
Admin				

- **Existence** tier: Principal knows the notebook exists but can't read it. Useful for "unlisted" shared notebooks.
- **Read+Write**: Can create and edit entries but not manage access or policies.
- **Admin**: Full control, including access control, retention policies, and deletion.

8. Audit Trail & Immutability Every operation is logged with: - **Actor** — Who performed the action (AuthorId) - **Action** — The operation type (WRITE, REVISE, SHARE, DELETE, etc.) - **Resource** — Which notebook or entry was affected - **Timestamp** — Wall-clock time (for auditing, not ordering) - **Status** — Success or failure, with error details - **Signature** — Cryptographic proof the log entry wasn't tampered with

Logs are **immutable**: Once written, they cannot be deleted or modified.

Platform Architecture

Cyber consists of three main components:

Backend (Rust)

- Core engine written in Rust (safety, performance, minimal dependencies)
- Five interconnected crates:
  - **notebook-core** — Entry types, cryptography, domain logic
  - **notebook-entropy** — Integration cost computation, clustering, coherence metrics
  - **notebook-store** — PostgreSQL persistence, Apache AGE graph queries
  - **notebook-server** — HTTP API (six operations + management endpoints)
  - **cli** — Command-line interface
- All stored data is cryptographically signed and immutable

Frontend (Blazor Server)

- Web UI for notebook/entry management, access control, auditing
- Server-rendered (tight security boundary, easier audit)
- Responsive design for desktop and tablet
- Keyboard shortcuts for power users

MCP Integration (Python)

- Model Context Protocol server for Claude Desktop
- Exposes all six operations as Claude tools
- JWT authentication (token-based)
- Ideal for AI-assisted knowledge creation and analysis

Who Should Use This Manual?

This manual is structured for **seven distinct user personas**, each with different goals and responsibilities:

1. **Knowledge Contributor** — Regular user creating and browsing notebook entries via MCP or UI
2. **Organization Administrator** — Setting up organizational structure, security clearances, and group membership
3. **Notebook Owner** — Creating notebooks, managing access, reviewing submissions, monitoring job processing
4. **Auditor/Compliance Officer** — Investigating security events, generating audit reports, ensuring compliance
5. **System Administrator** — Managing users, quotas, platform health, and global agent configuration
6. **ThinkerAgent Operator** — Deploying and managing AI processing workers that perform background jobs
7. **Cross-Organization Coordinator** — Managing knowledge sharing across organizational boundaries

Use this guide based on your role: - Just getting started? → Go to Chapter 3: Getting Started - **Know your role?** → Jump to the relevant chapter in Part II (Chapters 4-10) - **Need API details?** → Go to Part III: Reference (Chapters 11-16) - **Lost?** → Check Chapter 16: Glossary & Index or use your PDF reader's search feature

---

Key Design Principles

As you work with Cyber, you'll notice these principles reflected throughout:

1. **Security by Default** — Assume information is sensitive until proven otherwise. Information flows only to authenticated, authorized users.

2. **Causal Consistency Over Instant Consistency** — Accept that replicas may lag. Use causal positions, not wall-clock times, for ordering.

3. **Immutability as Feature** — Entries cannot be deleted; only new revisions. This preserves history and enables audit trails.

4. **Entropy Reflects Reality** — The platform doesn't mandate consensus; it measures and surfaces disagreement through integration costs.

5. **Federated, Not Centralized** — Users and organizations maintain cryptographic identity. No single point of failure or control.

What's Not Covered Here

This manual focuses on **user workflows and operational tasks**. For implementation details, architecture deep-dives, or extending the platform, see: - **Developer Guide** — backend/README.md - **Architecture Documentation** — docs/architecture/ - **Source Code** — github.com/cyber-project (Rust backend, Python client) - **Project Roadmap** — docs/project-plan.md

---

Moving Forward

You now understand the *why* and *what* of Cyber. The next chapter (Chapter 2: Security Model) goes deeper into classification levels, compartments, and access control rules. Then, Chapter 3: Getting Started walks you through your first login and interface orientation.

Ready to dive in? Turn to Chapter 2 →

Last updated: February 21, 2026 **Manual version:** 1.0.0 (Beta) **Platform version:** 2.1.0

Classification Levels

Cyber uses a **five-level classification hierarchy**. Each level represents increasing sensitivity and restricted distribution:

Level	Name	Typical Use	Distribution
1	`PUBLIC`	General company info, marketing, public research	No restrictions
2	`CONFIDENTIAL`	Internal memos, non-sensitive business data	Internal use only
3	`SECRET`	Strategic planning, customer data, technical designs	Need-to-know basis
4	`TOP_SECRET`	Military/national security, critical infrastructure	Severe impact if disclosed
5+	Custom	Organization-defined levels	Organization-defined

Dominance hierarchy:

`PUBLIC < CONFIDENTIAL < SECRET < TOP_SECRET < [Custom levels]`

A principal (user or group) cleared for a higher level can read all lower levels. A `TOP_SECRET` user can read `PUBLIC`, `CONFIDENTIAL`, and `SECRET`. A `CONFIDENTIAL` user cannot read `SECRET` or above.

Compartments (Domains)

Compartments are optional security categories that further restrict access *within* a classification level. They're used for:

- **Functional separation** — "Medical Research" vs. "Legal" vs. "Operations"
- **Project isolation** — "Project Alpha" vs. "Project Bravo"
- **Sensitive subjects** — "Executive Compensation", "Merger Negotiations", "Criminal Investigation"

Key rule: To access compartmented information, a user must be cleared for *both* the classification level AND the specific compartment.

Example compartments: - Medical Research - Strategic Planning - Infrastructure Operations - Customer PII - Executive / Confidential - International Operations

Organizations define their own compartment naming conventions. See Chapter 13: Security Reference for naming best practices.

Security Labels

Every principal, notebook, and entry has a **security label** combining level + compartments:

Format: `LEVEL / {compartment1, compartment2, ...}`

Examples:

Label	Meaning
`PUBLIC / {}`	No restrictions, anyone can access
`CONFIDENTIAL / {}`	Company-wide access only
`SECRET / {Medical Research}`	Medical researchers with SECRET clearance only
`TOP_SECRET / {Infrastructure, Operations}`	Users cleared for BOTH Infrastructure AND Operations compartments
`TOP_SECRET / {Executive}`	Senior leadership only

Empty compartment set (`{}`) means no compartment restrictions — anyone at that level can access.

Clearance Dominance

The security model uses **dominance rules** to determine whether a user can access information:

Definition: User clearance `C_user` **dominates** information label `L_info` if: 1. `C_user.level   L_info.level` AND 2. `L_info.compartments   C_user.compartments`

In other words: - User's level must be at least as high as the information's level, AND - User's compartments must be a *superset* of the information's compartments

Example calculations:

User: `TOP_SECRET / {Medical, Infrastructure, Strategic}`

Can access?	Information Label	Why?
	`PUBLIC / {}`	Level matches, no compartments needed
	`CONFIDENTIAL / {}`	Level matches, no compartments
	`SECRET / {Medical}`	Level matches, user has Medical compartment
	`TOP_SECRET / {Infrastructure}`	Level matches, user has Infrastructure compartment
	`TOP_SECRET / {Medical, Infrastructure}`	Level matches, user has both compartments
	`TOP_SECRET / {Medical, Infrastructure, Executive}`	User lacks Executive compartment
	`TOP_SECRET / {Finance}`	User lacks Finance compartment
	`SECRET / {Finance}`	User lacks Finance compartment (even though level is OK)

Information Flow Control

Bell-LaPadula's central rule: Information can only flow from a *lower* classification to *higher* classification (or same level).

What this means in practice:

1. **Read Rule ("Simple Security Property"):** A user can read information only if their clearance dominates the information's label.
2. **Write Rule ("*-Property" or "Confinement Property"):** A user can write to a notebook/entry only if the information's label dominates the user's clearance.

The Write Rule is tricky. Think of it this way:

- If you're cleared for `TOP_SECRET / {Medical}`, you should only write to notebooks labeled `TOP_SECRET / {Medical}` or higher
- You must NOT write to a `SECRET / {Medical}` notebook, because that would move information from your clearance level down to a lower level

- You CAN write to a `TOP_SECRET / {Medical, Infrastructure}` notebook, because you're adding information to a more restricted space

---

Information Flow Across Organizations

When subscribing to notebooks in *other* organizations, additional rules apply:

Cross-org subscription principle: Information flows only from lower to higher classification.

- Organization A's `CONFIDENTIAL` notebook can subscribe to Organization B's `PUBLIC` notebook
- Organization A's `PUBLIC` notebook cannot subscribe to Organization B's `SECRET` notebook
- Same clearance dominance rules apply (users must be cleared for subscribed content)

See Chapter 10: Cross-Organization Coordinator for subscription workflows.

Access Tiers Within a Classification

Once a user's clearance dominates an entry's security label, access is further restricted by **access tiers**, which control specific operations:

Tier	Meaning	Operations Allowed
Existence	Principal knows the resource exists	Can see resource in lists, but not read content
Read	Can read the resource	Read entries, browse catalog, search
Read+Write	Can modify the resource	Create entries, revise entries, update metadata
Admin	Full control	Manage access tiers, set policies, delete entries

Example: In a notebook labeled `SECRET / {Operations}`: - A user cleared for `SECRET / {Operations}` might have "Read" tier (can view, but not edit) - The notebook owner has "Admin" tier (full control) - A contractor might have "Existence" tier (knows it exists, but can't read)

Access tiers are **separate from** classification levels. You can dominate the classification but still lack write access.

Clearance Calculation in Practice

When you attempt an operation (read, write, admin), Cyber performs these checks:

```
1. Is the user authenticated?
   No → Deny (unauthenticated access)

2. Does user's clearance dominate the notebook's classification?
   No → Deny (user not cleared for this content)

3. Does user's clearance dominate the specific entry's classification?
   (Entries can have more restrictive labels than their notebook)
   No → Deny (user not cleared for this specific entry)

4. Does user's access tier allow this operation?
   - Reading? Need "Read" or higher
   - Writing? Need "Read+Write" or higher
   - Admin? Need "Admin" tier
   No → Deny (insufficient permissions)

5. Pass all checks → Allow
```

If any check fails, the operation is denied and logged.

Practical Examples

Scenario 1: Medical Research Organization **Organization:** Health.Corp

Users: - Dr. Alice: `TOP_SECRET / {Medical Research, Operations}` - Nurse Bob: `CONFIDENTIAL / {Medical Research}` - Accountant Carol: `CONFIDENTIAL / {Finance}`

Notebooks: - "Research Phase 3 Trials" — `TOP_SECRET / {Medical Research}` - "Patient Demographics" — `SECRET / {Medical Research}` - "Operations Budget" — `CONFIDENTIAL / {Finance}`

Who can access what?

User	Research Phase 3	Patient Demographics	Operations Budget
Dr. Alice	(dominates)	(dominates)	(lacks Finance)
Nurse Bob	(level too low)	(dominates)	(lacks Finance)
Accountant Carol	(level too low)	(level too low)	(dominates)

Scenario 2: Multi-Project Company **Company:** TechCorp

User: Engineer Eve (clearance: `SECRET / {ProjectAlpha, ProjectBeta, Infrastructure}`)

Notebooks: - "ProjectAlpha Source Code" — `SECRET / {ProjectAlpha}` - "ProjectAlpha + Beta Integration" — `SECRET / {ProjectAlpha, ProjectBeta}` - "ProjectGamma Skunkworks" — `TOP_SECRET / {ProjectGamma}` - "Infrastructure Hardening" — `SECRET / {Infrastructure}`

Who can access what?

Notebook	Can Eve Access?	Why?
ProjectAlpha		Eve has ProjectAlpha compartment
ProjectAlpha + Beta		Eve has both compartments
ProjectGamma		Eve lacks ProjectGamma clearance
Infrastructure		Eve has Infrastructure compartment

Eve tries to write to each:

Notebook	Can Eve Write?	Why?
ProjectAlpha		Eve's clearance dominates (same level, same compartments)
ProjectAlpha + Beta		Notebook is more restricted (requires Beta, which Eve has, but write rule: you can only write if your clearance is notebook's, not <)
Infrastructure		Eve's clearance dominates

Note: The write rule prevents "downgrading" information. If Eve writes content cleared for `SECRET / {ProjectAlpha, ProjectBeta}` to a notebook labeled `SECRET / {ProjectAlpha}`, she's moving restricted info to a less restricted space.

Clearance Dominance Rules (Reference)

For quick lookup, here's the formal definition:

```
Clearance C dominates Label L if:
  C.level   L.level AND
  L.compartments   C.compartments

Examples:
- TOP_SECRET / {A, B, C} dominates TOP_SECRET / {A, B}
- TOP_SECRET / {A, B} dominates TOP_SECRET / {A, B, C}
- SECRET / {A, B} dominates SECRET / {A}
- TOP_SECRET / {} dominates SECRET / {A}   (no compartment restrictions)
- PUBLIC / {A} dominates CONFIDENTIAL / {}   (level too low)
```

Common Security Decisions

Decision: Should this notebook be classified? Use this matrix to determine the appropriate classification level:

Risk of Disclosure	Impact	Level
None	Public knowledge	`PUBLIC`
Low	Minor embarrassment	`CONFIDENTIAL`
Medium	Competitive disadvantage	`SECRET`
High	Severe impact (financial, legal, safety)	`TOP_SECRET`

Decision: Do we need compartments? Create compartments if: - Different audiences need different subsets of information - Projects or teams are isolated - Sensitive subjects need extra restriction - Regulatory requirements mandate it (HIPAA, ITAR, etc.)

Don't create compartments for: - Purely organizational purposes (use notebook hierarchies instead) - Temporary groupings (delete them, not archive) - Redundant categories (avoid nested compartments like `{Medical-Research-Phase-1}`)

Decision: What clearance should a user have? Principle: Users should have the **minimum clearance necessary** to do their job.

- Don't grant `TOP_SECRET` if `SECRET` is sufficient
- Don't grant broad compartments; grant only those needed
- Review clearances quarterly; remove unnecessary ones
- Document the business justification for each clearance

Troubleshooting Access Denials

You see: "Access Denied" or "Not Authorized"

Use this decision tree:

```
1. Am I logged in?
   No → Log in first

2. Am I trying to access my own notebook?
   No → Skip to step 3
   Yes → Check notebook classification.
          Are you (the owner) still cleared for your own notebook?

3. What is the notebook's classification label?
   Ask the notebook owner or check /notebooks page

4. What is my clearance label?
```

```
   Check your user profile (/profile)

5. Does my clearance dominate the notebook's label?
   Use the dominance rule above
   No → Request clearance upgrade from your org admin

6. What operation am I trying (read/write/admin)?
   Check my access tier for this notebook
   No → Request access tier upgrade from notebook owner

7. Still blocked?
   Contact your security officer or notebook owner
```

Best Practices

1. **Classify conservatively** — Classify information at the lowest level that protects it. Over-classification reduces information sharing and creates compliance burdens.
2. **Compartments are for separation, not granularity** — Use a small number of compartments. If you have more than 10 per organization, reconsider your strategy.
3. **Review clearances regularly** — Users' roles change. Audit clearances quarterly and remove unnecessary ones.
4. **Log access violations** — Cyber automatically logs all access denials. Review them monthly to catch policy issues or attacks.
5. **Use access tiers for least privilege** — Don't grant "Admin" to everyone. Use "Read+Write" by default, "Admin" only for owners.
6. **Communicate classification clearly** — Every notebook and entry displays its classification. Users should understand why information is restricted.
7. **Test before deploying** — Create test notebooks with different classifications. Verify that access rules work as expected before moving to production.

Next Steps

Now that you understand the security model, you're ready to:

1. **Chapter 3: Getting Started** — First login and basic orientation
2. **Chapter 13: Security Reference** — Deep dive into compartment naming, classification examples, and decision trees
3. **Jump to your role** — Part II has persona-specific guides for different jobs

---

Last updated: February 21, 2026 **Manual version:** 1.0.0 (Beta) **Platform version:** 2.1.0 **Security model:** Bell-LaPadula (NIST SP 800-95)

Your organization administrator will provide you with: - **Instance URL** — Where to access Cyber (e.g., `https://cyber.company.com`) - **Authentication method** — OIDC integration, SAML, or custom (depends on your org) - **Initial clearance level** — Your starting security clearance (e.g., `CONFIDENTIAL / {}`)

If you don't have these, contact your organization's Cyber administrator.

Step 2: First Login

1. Navigate to your Cyber instance URL in a web browser
2. Click **"Sign In"** or **"Create Account"**
3. Follow your organization's authentication flow:
   - **OIDC:** Use your company SSO (Google, Okta, Azure AD)
   - **SAML:** Authenticate through your enterprise identity provider

   - **Email/Password:** Verify your email address
4. On first login, you'll be prompted to generate a cryptographic key pair

Step 3: Cryptographic Key Generation Cyber uses **Ed25519 keys** for signing all operations. On first login:

1. You'll see a dialog: **"Generate Your Signing Key"**
2. Click **"Generate New Key"** — The browser will create a public/private key pair locally
3. Your **private key will be saved in browser storage** (encrypted with your password)
4. Your **public key is registered with the server** and used to verify your identity

Important security notes: - Your private key never leaves your browser (unless you export it) - Lose your key? You'll need to generate a new one (old entries remain but you can't sign new ones) - Backup your key via the **Profile → Security** page if you want to restore it on another device

Step 4: Set Your Profile After key generation, you'll be prompted to complete your profile:

- **Full Name** — Display name for audit logs and collaboration
- **Email** — Contact info for notifications and password resets
- **Avatar** — Optional profile picture
- **Organization** — Which organization you belong to
- **Department/Team** — For group membership and organization charts

Once completed, you'll see the **Dashboard**.

The Dashboard

The Dashboard is your home page after logging in. It provides an at-a-glance view of your activity and the platform's health.

Dashboard Sections

```
Dashboard                                      [User Menu]


  System Status
• Notebooks: 42 total, 8 new this week
• Entries: 2,341 total, 156 added today
• Pending Jobs: 3 DISTILL_CLAIMS, 2 COMPARE_CLAIMS
• Health:  All systems nominal

  Your Recent Activity
• Jan 21 - You revised "API Architecture" entry
• Jan 20 - You created 3 new entries in Q1 Planning
• Jan 19 - Project Oversight group added you as member

  Your Notebooks (Quick Access)

  Name                Entries    Access

  Q1 Planning         45         Admin
  R&D Notes           128        Read+W
  Strategic Roadmap   12         Read


  Security Events (Last 7 days)
• 2 Access Denials - Jan 20, IP 192.168.1.50
• 0 Failed Auth Attempts
• 1 Clearance Change - Jan 19, added SECRET level

  Recommended Actions
```

- `Set up MCP access for Claude Desktop`
- `Review pending group invitations (1 pending)`

Key metrics:

Widget	Shows You
System Status	Platform health, total notebooks/entries, pending background jobs
Recent Activity	Actions you took (created/revised entries, group changes, etc.)
Your Notebooks	Quick access to notebooks you own or have access to
Security Events	Access denials, login failures, clearance changes
Recommended Actions	Setup tasks, invitations, pending reviews

Navigation Sidebar On the left side of every page:

```
Cyber (logo)

 Dashboard
 Notebooks
 Entries
 Explore
 Search
[Divider]
 Settings
 Profile
 Security
 Audit Log
[Divider]
 Admin Panel (if you're an admin)
```

Understanding Your Permissions

On your **Profile** page (`/profile`), you'll see three key pieces of information:

1. Your Clearance

```
 Your Clearance Level

Current: CONFIDENTIAL / {Strategic Planning, Operations}

What this means:
   You can read any PUBLIC or CONFIDENTIAL notebook
   You can read CONFIDENTIAL entries in Strategic Planning and Operations
   You cannot access SECRET, TOP_SECRET, or other compartments

Request a clearance upgrade: [Contact Admin]
```

Your clearance determines what information you can *read*. If you need access to a more-restricted notebook, contact your organization administrator.

2. Your Group Memberships

```
 Groups

You are a member of:
  • Engineering Team (Member role)
  • Project Alpha (Member role)
  • Executive Council (Admin role)  ← You can add members to this group
```

Groups affect: - Which notebooks you automatically have access to - Your administrative responsibilities - Your audit permissions (group admins see group-related events)

3. Your Authentication Keys

```
 Signing Keys

Active: Ed25519 public key 0x8a2f... (created Jan 18, 2026)
Backup keys: None

Export private key (for backup/restore): [Download]
```

Manage your cryptographic keys here. You need at least one active key to sign new entries.

Creating Your First Notebook

Now that you're set up, let's create your first notebook:

Step 1: Go to Notebooks Page

1. Click **Notebooks** in the left sidebar
2. Click **"+ New Notebook"** button
3. You'll see a form:

```
Create New Notebook



Name *
[Text field: "Q1 Project Planning"]

Description
[Large text field: "Central hub for Q1 priorities, milestones, and team coordination"]

Owner Group *
[Dropdown: "Select a group..."]
    → Engineering Team
    → Project Oversight
    → Strategic Planning

Classification Level (Advanced)
[Dropdown: "CONFIDENTIAL"]  ← Inherited from owner group

Compartments (Optional)
[Tag field: + Add compartments...]
    Examples: Strategic Planning, Medical Research, etc.

[Create Notebook]  [Cancel]
```

Step 2: Fill in Details

- **Name:** Concise, clear (e.g., "Q1 Planning", "Patient Records", "R&D Roadmap")
- **Description:** 1-2 sentences explaining the notebook's purpose
- **Owner Group:** The group responsible for this notebook
  - Only group admins can manage the notebook
  - All group members get at least "Read" access
- **Classification:** Usually inherited from group, but can be more restrictive
- **Compartments:** Optional security categories (e.g., if it contains sensitive personal data)

Step 3: Create & Configure Access Click **"Create Notebook"**. You'll be redirected to the notebook's **Settings** page:

```
Notebook: Q1 Project Planning


  Entry Feed    Settings    Access Control    Statistics

[Settings Tab Active]

Classification: CONFIDENTIAL / {}
Entry Retention: 7 years (default)
Status: Active

Access Control


Current Members:

Name           | Role    | Tier       | Actions

Engineering   | Group   | Read+Write | Remove
(4 members)   |         |            |

You (Jane)    | Owner   | Admin      | (You)

[+ Add User or Group]
```

By default: - Your owner group has **Read+Write** access - You have **Admin** access - Others can be added individually

Step 4: Add Collaborators (Optional) To give other users access:

1. Click **"+ Add User or Group"**
2. Search for a user or group by name
3. Select the **access tier** (Existence / Read / Read+Write / Admin)
4. Click **"Add"**

The user will see the notebook in their **Notebooks** page and can start reading/contributing.

Reading Your First Entry

Once a notebook exists, you can start reading entries. Here's the **Entry Feed** view:

```
Q1 Project Planning


Entry Feed | Settings | Access Control | Statistics

Filter & Search:
[Topic dropdown: All] [Status: All] [Friction: All] [Search box: ___________]

  Pinned Entries (0)

Recent Entries:


[Entry Card]
  Title: "Q1 Goals and Priorities"
Author: Jane Smith (Jan 22, 2026)
Integration Status:  Integrated (low friction)
Topic: organization/planning/goals
References: 3 entries

Quick view: [Read] [History] [Compare]
```

```
[Entry Card]
 Title: "Team Resource Allocation"
Author: Bob Johnson (Jan 21, 2026)
Integration Status:  Probation (calculating friction)
Topic: organization/planning/resources
References: 2 entries

Quick view: [Read] [History] [Compare]
```

To read an entry:

1. Click on an entry card
2. The full entry opens in a side panel:

```
  Q1 Goals and Priorities                         [Close ×]


Content:


# Q1 Goals and Priorities

For Q1 2026, we're focusing on three strategic pillars:

1. **Customer Experience** - Reduce support ticket response time
   by 50% and increase satisfaction scores above 4.5/5.0

2. **Infrastructure Reliability** - Zero critical incidents,
   99.99% uptime SLA

3. **Team Development** - Complete certifications for 100% of
   engineering team

---

Entry Metadata:


Author:         Jane Smith
Created:        Jan 22, 2026, 10:30 AM
Position:       42 (causal ordering)
Integration:    Integrated
Friction Score: 0.21 (low - well aligned with existing entries)
Topic:          organization/planning/goals
References:     [Q1 Budget] [Engineering Roadmap] [Team Charter]

[Revise] [Compare with Other Versions] [View History]
```

Key elements:

Element	What It Means
Position	Causal order (42 = 42nd entry in this notebook)
Integration	Status: probation (new), integrated (stable), or contested (contradictory)
Friction Score	0-10: How much this entry disrupts existing knowledge (0 = perfectly aligned, 10 = major disagreement)
Topic	Hierarchical classification (e.g., org/planning/goals)
References	Related entries this one links to

Searching and Browsing

Full-Text Search Use the search box at the top of any page:

1. Type your query (e.g., “budget allocation”)
2. Press Enter or click **Search**
3. Results appear sorted by relevance

Search syntax:

```
Query Type              | Example

Simple keyword          | budget
Exact phrase            | "Q1 budget"
Author                  | author:Jane
Classification level    | level:SECRET
Topic filter            | topic:planning
Friction threshold      | friction:>5
Combination             | "budget" author:Jane friction:<3
```

Browsing by Topic

1. Go to **Explore** in the sidebar
2. You’ll see a hierarchical topic tree:

```
 Explore Notebooks



You have access to X notebooks across these topics:

 organization/
    planning/
       goals
       budget
       roadmap
    operations/
       incidents
       runbooks

 projects/
    alpha/
       architecture
       schedule
```

Click any topic to see all entries under that category.

Interface Overview

Key Pages

Page	URL	Purpose
Dashboard	`/`	Home page, system status, recommendations
Notebooks	`/notebooks`	List of all notebooks you have access to
Entries	`/entries`	Global entry search and filtering
Explore	`/explore`	Browse by topic hierarchy
Search	`/search`	Advanced full-text search
Profile	`/profile`	Your account, clearance, groups, keys
Settings	`/settings`	Personal preferences, notifications, API tokens
Admin Panel	`/admin`	User management, audit logs, system config (admins only)

Keyboard Shortcuts

Shortcut	Action
`?`	Show this help menu
`/`	Focus search box
`n`	New entry/notebook
`e`	Enter/exit edit mode
`s`	Save
`Esc`	Close modals, exit edit
`g d`	Go to Dashboard
`g n`	Go to Notebooks
`g e`	Go to Entries

(Disabled in text input fields to avoid conflicts)

Generating API Tokens

If you plan to use the **MCP integration** or **REST API** programmatically:

Step 1: Go to Settings

1. Click your avatar in the top-right corner
2. Select **Settings → API Tokens**

Step 2: Create a New Token

```
API Tokens


Active Tokens:
(none yet)

[+ Generate New Token]
```

Click **"+ Generate New Token"**:

```
Create API Token


Token Name: [Claude Desktop MCP]

Expiration:  Never   1 Month   90 Days   1 Year

Scopes (what this token can do):
 Read notebooks and entries
 Write and revise entries
 Manage access control
 View audit logs
 Delete entries
 Administer users and groups

[Generate Token]
```

Step 3: Copy and Store Securely Once generated, you'll see:

```
 Token created!

CYBER_TOKEN=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...

 Save this token somewhere safe. You won't see it again.
```

```
   If you lose it, generate a new one.

[Copy to Clipboard] [Done]
```

Security notes: - Treat this token like a password - Use environment variables to store it (not in code) - Rotate tokens yearly - Delete tokens you no longer use

---

Accessibility Features

Cyber is designed for accessibility:

Feature	Use Case
High contrast mode	Settings → Appearance → High Contrast
Large fonts	Settings → Appearance → Font Size
Dark mode	Settings → Appearance → Dark Mode
Screen reader support	All UI elements have ARIA labels
Keyboard navigation	Use `Tab` to navigate, `Enter` to activate
Text-to-speech	[Select text and right-click "Read Aloud"]

Next Steps

Congratulations! You've completed the basic setup. Now it's time to get to work:

- **Creating entries?** → Go to Part II, Your Role
- **Setting up MCP?** → Workflow: MCP Setup for Knowledge Contributors
- **Exploring security?** → Chapter 2: Security Model
- **Need help?** → Chapter 15: Troubleshooting

Last updated: February 21, 2026 **Manual version:** 1.0.0 (Beta) **Platform version:** 2.1.0

- Create well-structured entries in assigned notebooks
- Organize content with clear topics and references
- Discover and learn from existing knowledge
- Revise and improve entries based on feedback
- Monitor changes to stay informed on evolving topics
- Collaborate with other contributors through references and causal linking

Required Permissions: - At least "Read" access to one or more notebooks - "Read+Write" access to contribute new entries - Your organizational clearance (inherited from your role)

Typical Workflows: 5 core workflows in this chapter

Workflow 1: Creating and Organizing Entries

Overview

Learn how to create a new entry in a notebook, structure it with topics, and link it to related entries. This is the foundational workflow for all contributors.

Use case: You have new knowledge (research findings, meeting notes, architectural decisions) that needs to be recorded in your team's notebook.

Related workflows: - Managing Revisions — Update entries after creation - Browsing Knowledge — Find related entries to reference - Observing Changes — Track what others add

Prerequisites

- □ Cyber account created and authenticated
- □ At least "Read+Write" access to a notebook
- □ Understanding of your notebook's topic structure
- □ Content ready to enter (notes, document, research)

Step-by-Step Instructions

Step 1: Navigate to Your Notebook **UI Path:** Left sidebar → Notebooks → Select notebook name

1. Click **Notebooks** in the left sidebar
2. You'll see a list of notebooks you have access to
3. Click the notebook where you want to add an entry
4. You'll see the **Entry Feed** with existing entries

Example:

```
Your Notebooks

  Q1 Planning (Read+Write) ← Click here
  R&D Notes (Read+Write)
  Strategic Roadmap (Read only)
```

Step 2: Click "New Entry" Once in the notebook, look for the **"+ New Entry"** button:

```
Q1 Planning


[+ New Entry]  [Filters]  [Search box]

Entry Feed:
(list of existing entries)
```

Click **"+ New Entry"** to open the entry creation form.

Step 3: Fill in Entry Details You'll see a form with these fields:

```
Create New Entry


Title *
[Engineering Roadmap Q1 2026]

Topic *
[organization/engineering/roadmap]
  ↓ (click to browse topic hierarchy)

Content *
[Large text editor - use Markdown for formatting]

References (Optional)
[+] Add entry references...
  Search: [____________]
  [Quarterly Goals] [Q1 Budget] [Team Charter]

[Create Entry]  [Preview]  [Save as Draft]  [Cancel]
```

Field Explanations:

Field	Required	Guidelines
Title	Yes	Clear, specific (e.g., "Engineering Roadmap Q1 2026", not "Stuff")

Field	Required	Guidelines
Topic	Yes	Hierarchical path (e.g., org/engineering/roadmap). Start with org, team, or project name.
Content	Yes	Supports Markdown formatting (headers, lists, code blocks, links)
References	No	Link to related entries for context and cross-referencing

Step 4: Write Your Content The content editor supports **Markdown formatting**:

```
# Engineering Roadmap Q1 2026

## Overview
This quarter we're focusing on infrastructure modernization.

## Key Initiatives

1. **Kubernetes Migration**
   - Timeline: Jan - Mar 2026
   - Team: DevOps + SRE
   - Status: In Progress

2. **API v2 Release**
   - Timeline: Feb - Mar 2026
   - Team: Backend Engineering
   - Status: Design phase

## Success Criteria
- [ ] All services containerized
- [ ] Zero-downtime deployments
- [ ] < 100ms API latency (p99)

---

**See also:** [Quarterly Goals], [Team Charter], [Infrastructure Budget]
```

Pro tips: - Use headers (`# Big Title`, `## Smaller Heading`) to structure content - Use bullet points and numbered lists for clarity - Include status indicators (Done, In Progress, Blocked) - Add checkboxes for tracking tasks

Step 5: Add References (Optional but Recommended) References link your entry to related entries, creating a knowledge graph:

1. Click **"[+] Add References"** at the bottom
2. Search for entries by title or topic:

   ```
   Search references...

     [quarterly goals_______________]


   Results:
     Quarterly Goals (engineering/planning)
     Q1 OKRs Overview (organization/goals)
     Quarterly Budget Review (finance/budgets)
   ```

3. Check the entries you want to reference
4. Click **"Add Selected"**

What references do: - Create bidirectional links (both entries reference each other) - Help Cyber measure entry coherence (integration cost) - Allow readers to discover related content - Build the knowledge graph structure

Step 6: Preview (Optional) Click **"Preview"** to see how your entry will look:

```
Engineering Roadmap Q1 2026


Author: You (Jane Smith)
Topic: organization/engineering/roadmap
References: 3 entries linked

## Overview
This quarter we're focusing on infrastructure modernization.
...
```

Step 7: Create Entry Click **"Create Entry"**. You'll see:

```
  Entry created successfully!

Entry ID: entry_abc123
Position: 127
Integration Status: Probation (calculating friction)

[View Entry] [View in Notebook] [Create Another]
```

What happens next: 1. Your entry is signed with your cryptographic key (proof of authorship) 2. Background jobs analyze it for integration cost 3. Entry goes into "Probation" status while being analyzed 4. Within 1-5 minutes, it stabilizes to "Integrated" or "Contested" status

Verification

Confirm your entry was created successfully:

- ☐ Entry appears at the bottom of your notebook's entry feed
- ☐ Title and topic are correct
- ☐ Content displays properly (Markdown formatted)
- ☐ References are linked correctly
- ☐ Your name appears as the author
- ☐ Timestamp shows current date/time
- ☐ Integration status shows "Probation" (will change to "Integrated")

Tips & Tricks

Shortcut: Use MCP Integration If you have MCP set up (see Workflow 1 from Chapter 4), you can create entries via Claude:

```
Claude: Create a new entry in the Q1 Planning notebook:
  Title: Engineering Roadmap Q1 2026
  Topic: organization/engineering/roadmap
  Content: [your content]
```

Claude will create the entry and sign it automatically.

Batch Import (Advanced) For importing many entries at once, use the CLI:

```
cyber write --notebook-id nb_xyz \
  --title "My Entry" \
  --topic "org/team/subject" \
  --content "$(cat file.md)" \
  --references entry_1,entry_2
```

Draft Saving Click **"Save as Draft"** to save without creating yet. Drafts are stored locally in your browser until you're ready to publish.

Structured Data For technical entries, use code blocks:

~~~
## Configuration

```yaml
database:
  host: db.prod.internal
  port: 5432
  replica_count: 3
```

```json
{
  "api_version": "v2",
  "deprecation_date": "2026-06-01"
}
```
~~~

Next Steps

After creating your entry: - Browse and discover other entries to build context - Manage revisions if you need to update your entry - Observe changes to see how others respond

---

Workflow 2: Browsing and Discovering Knowledge

Overview

Learn how to search, filter, and navigate existing entries to find the information you need and understand how it connects to your work.

Use case: You're starting a new project and need to understand existing decisions, architecture, or past experiences on similar topics.

Related workflows: - Creating Entries — Add new entries informed by what you discover - Observing Changes — Monitor knowledge you're interested in

Prerequisites

- ☐ Cyber account and at least "Read" access to notebooks
- ☐ Understanding of your organization's topic structure

Step-by-Step Instructions

Method 1: Full-Text Search **Search box location:** Top of any page (keyboard shortcut: `/`)

1. Click the search box at the top
2. Type your query:

   ```
   Search box
   [Kubernetes migration_____]
   ```

3. Press Enter or click Search
4. Results appear ranked by relevance:

   ```
   Search Results for "Kubernetes migration"


   [Relevance:       ] Kubernetes Migration Plan
     Topic: organization/infrastructure/migration
     Author: DevOps Team
     Created: Jan 15, 2026
     "We're planning a phased migration to Kubernetes..."

   [Relevance:       ] K8s Security Considerations
   ```

```
   Topic: organization/infrastructure/security
   Author: Security Team
   Created: Jan 10, 2026
```

Search syntax:

```
Exact phrase:     "Kubernetes migration"
Author filter:    author:Alice
Topic filter:     topic:infrastructure
Classification:   level:SECRET
Friction range:   friction:>5 (high friction/controversial)
```

Method 2: Topic Hierarchy Browse **Navigate to:** Sidebar → Explore

1. Click **"Explore"** in the sidebar
2. You'll see a hierarchical topic tree:

```
 Explore


You have access to 42 notebooks

 organization/
    engineering/
       backend/
          Database Migrations
          API Architecture
          Performance Optimization
       infrastructure/
          cloud/
             Kubernetes Migration
             Multi-cloud Strategy
    operations/
       incidents/
          2026-02 Outage Report
```

3. Click any topic to see all entries under it
4. Entries are listed with metadata:

```
Topic: organization/engineering/infrastructure/cloud


[Entry] Kubernetes Migration Plan
   Author: DevOps Team  |  Created: Jan 15, 2026
   Integration:  Integrated (stable)
   Friction: 0.34 (low - well aligned)
   References: 3 entries
   [Read] [History] [Compare]

[Entry] Multi-cloud Strategy
   Author: Infrastructure Team  |  Created: Jan 10, 2026
   Integration:  Probation (still calculating)
   Friction: 2.1 (medium - some disagreement)
   References: 2 entries
```

Method 3: Browse Your Notebook's Entries **Navigate to:** Notebooks → Select notebook → Entry Feed

1. Go to **Notebooks** in the sidebar
2. Click a specific notebook
3. You'll see the **Entry Feed** with filters:

```
Q1 Planning
```

```
[Filter: Topic ] [Filter: Status ] [Filter: Friction ]

[Topic dropdown]
 All Topics
 organization/planning
 organization/planning/goals
 organization/planning/budget

[Status dropdown]
 All Statuses
 Integrated
 Probation
 Contested

[Friction dropdown]
 All Friction
 Low (0-2)
 Medium (2-5)
 High (5-10)
```

4. Select filters to narrow results
5. Entries appear sorted by **newest first** (or selected filter)

Step 4: Read an Entry Click any entry to open the full view:

```
Kubernetes Migration Plan


## Overview

We're planning a phased migration to Kubernetes over
the next three months...

[Full content displayed in readable format]

---

Metadata:


Author:         DevOps Team (Alice Chen)
Created:        Jan 15, 2026, 10:30 AM
Position:       127 (causal order)
Integration:      Integrated (stable)
Friction:       0.34 (low - well aligned with existing entries)
Topic:          organization/engineering/infrastructure/cloud
References:     → Multi-cloud Strategy
                → Q1 Budget Plan
                → Team Charter

This entry is referenced by:
                ← Infrastructure Roadmap
                ← Jan All-Hands Notes

[Related Entries] [View History] [Compare Versions] [Discussion]
```

Step 5: Understand Integration Status Each entry shows its **integration status**:

Status	Meaning	What to Do
Integrated	Stable, well-aligned with other knowledge	Safe to rely on, reference in your work
Probation	New, still being analyzed for coherence	Wait a few minutes for final status, check back
Contested	High friction, contradicts other entries	Investigate disagreement, discuss with authors

High friction doesn't mean wrong — it might mean: - This is a novel/innovative idea (not yet mainstream) - Legitimate disagreement between approaches - Outdated information vs. newer insights - Different contexts (what works for one team may not work for another)

Verification

Confirm you're effectively discovering knowledge:

- ☐ Found at least one entry related to your current project
- ☐ Used at least two discovery methods (search, topic browse, notebook feed)
- ☐ Understood the relationship between entries (references, friction)
- ☐ Noted entries with high friction for follow-up discussion
- ☐ Bookmarked or noted entry IDs for later reference

Tips & Tricks

Use Friction Filtering for Learning

- **Low friction (0-2):** Established best practices, safe to follow
- **Medium friction (2-5):** Evolving approaches, worth understanding context
- **High friction (5-10):** Controversial or novel ideas, engage with authors

Follow Related Entries When reading an entry, click **"Related Entries"** to see: - Entries it references (what it builds on) - Entries that reference it (what builds on this) - Entries on the same topic

This creates a **knowledge exploration path**.

Watch Authors If you find entries by great authors, click their name to see other entries they've created. Good contributors are gold mines of knowledge.

Use Causal Positions Each entry has a **position number** (e.g., Position 127). Lower numbers = older, higher = newer within that notebook. This helps understand timeline of decisions.

Next Steps

After discovering knowledge: - Create an entry building on what you've learned - Discuss high-friction entries with authors - Reference the entries you found in your own work

---

Workflow 3: Searching Across Notebooks

Overview

Search simultaneously across multiple notebooks and organizations to find knowledge regardless of where it lives.

Use case: You're investigating a cross-cutting concern (e.g., security, compliance, architecture patterns) that spans multiple teams.

Prerequisites

- ☐ At least "Read" access to 2+ notebooks
- ☐ Clear understanding of what you're searching for

Step-by-Step Instructions

The **Global Search** is accessible from anywhere:

1. Press `/` (forward slash) on your keyboard
2. Or click the Search icon in the sidebar
3. Enter your query:

   ```
   Global Search


   [Encryption standards____________]

     Searching across all accessible notebooks...
   ```

4. Results appear with filters:

   ```
   Results for "encryption standards" (42 matches)


   [Notebook: All  ] [Author: All  ] [Date: All  ]

   [Relevance      ] Encryption Standards v2
     From: Operations/Security
     Author: Security Team
     Created: Jan 2026

   [Relevance      ] TLS Configuration Guide
     From: Engineering/Infrastructure
     Author: DevOps Team
     Created: Dec 2025
   ```

Advanced Filters: - Filter by notebook, author, date range - Sort by relevance or date - View entry counts per notebook

Verification

☐ Found entries across multiple notebooks
☐ Used filters to narrow results effectively
☐ Compared approaches between teams
☐ Created an entry synthesizing findings

Workflow 4: Managing Revisions

Overview

Learn how to update entries over time. Cyber uses immutable revisions—you don't edit entries, you create new versions that supersede old ones.

Use case: You created an entry about a project roadmap, and it needs updating after a planning meeting. You revise it, creating a new version.

Related workflows: - Creating Entries — Your initial entry - Observing Changes — Track revisions others make

Prerequisites

☐ "Read+Write" access to the notebook containing the entry
☐ The entry you want to revise
☐ Clear understanding of what needs to change

Step-by-Step Instructions

Step 1: Find the Entry to Revise Navigate to the entry (via notebook, search, or browse).

Click “**[Revise]**” button:

```
Engineering Roadmap Q1 2026


[Read] [History] [Revise] [Compare]
                 ↑ Click here
```

Step 2: Create a Revision A new form appears with the **previous version’s content pre-filled**:

```
Revise Entry


Original Entry ID: entry_abc123
Revision Reason: [Updated after Jan planning meeting_____]

Content *
[Previous content pre-populated...]

[Update Entry] [Preview] [Cancel]
```

Step 3: Make Your Changes Edit the content as needed:

```
Original:
## Key Initiatives
1. Kubernetes Migration
2. API v2 Release

Updated:
## Key Initiatives
1. Kubernetes Migration (timeline: Jan-Mar → Feb-Apr)
2. API v2 Release
3. Database Optimization (new initiative)
```

Step 4: Add a Reason In the **“Revision Reason”** field, explain why you’re revising:

```
Revision Reason Examples:
- "Updated after Jan 15 planning meeting"
- "Fixed typo in timeline"
- "Added new Q1 initiatives approved by leadership"
- "Corrected infrastructure budget numbers"
```

Good reasons help readers understand the change context.

Step 5: Submit Revision Click **“Update Entry”**:

```
  Revision created successfully!

Original Entry:    entry_abc123, Position 127
New Revision:      entry_def456, Position 128
Reason:            Updated after Jan planning meeting

You can:
  [View New Revision] [View History] [Compare Versions]
```

What Happens to the Old Version?

- Old version is preserved forever (immutable)
- New revision shows as current in the entry feed
- Readers see the new version by default
- History shows all revisions (with reasons)

- You can compare old vs. new side-by-side

Verification

Confirm your revision:

☐ New version appears in the entry feed
☐ Revision reason is recorded
☐ History shows both old and new versions
☐ Changes are visible in the new version
☐ Revision count increments

Tips & Tricks

View Entry History Click **“[History]”** to see all versions:

```
Entry History: Engineering Roadmap Q1 2026


Version 3 (Current) - Position 129
  Author: Jane Smith
  Date: Jan 22, 2026, 2:30 PM
  Reason: Added database optimization initiative
  [View] [Compare with v2]

Version 2 - Position 128
  Author: Jane Smith
  Date: Jan 15, 2026, 10:30 AM
  Reason: Updated timeline after planning meeting
  [View] [Compare with v1]

Version 1 (Original) - Position 127
  Author: Jane Smith
  Date: Jan 10, 2026, 9:00 AM
  Reason: (original creation)
  [View]
```

Compare Versions Click **“[Compare]”** to see differences:

```
Comparison: v1 vs. v3


- Kubernetes Migration (timeline: Jan-Mar)
+ Kubernetes Migration (timeline: Feb-Apr)

  API v2 Release

+ Database Optimization (new initiative)
```

Revision Frequency

- **Small fixes** (typos, formatting): Revise immediately
- **Major changes** (scope, timeline, approach): Coordinate with stakeholders first
- **Multiple small changes**: Batch them into one revision with clear reason

Next Steps

After revising: - Notify stakeholders if it’s an important change - Check if dependent entries need updating - Monitor discussion/comments on the revision

---

Workflow 5: Observing Changes

Overview

Learn how to track changes to notebooks you care about, staying informed without manually checking repeatedly.

Use case: You're implementing a feature based on an architectural entry, and want to know if requirements change.

Related workflows: - Browsing Knowledge — Find entries to observe - Creating Entries — Contribute your own changes

Prerequisites

- □ At least "Read" access to a notebook
- □ Specific notebook or entry you want to monitor

Step-by-Step Instructions

Method 1: Watch a Notebook for Changes **Via UI:**

1. Go to a notebook (Notebooks → Select notebook)
2. Click **"Watch"** or **"Subscribe"** button (location varies)
3. Select notification frequency:

```
Watch Notebook


Q1 Planning

Notify me of:
  New entries
  Revisions to existing entries
  Comments/discussions
  Integration status changes

Frequency:
  Immediately
  Daily digest
  Weekly summary
  Never (just view history)

 [Save Preferences]
```

4. You'll receive notifications matching your preferences

Method 2: Use the OBSERVE Operation (Advanced) If using MCP or REST API, use the **OBSERVE** operation:

```
curl -X GET http://localhost:8000/observe \
  -H "Authorization: Bearer TOKEN" \
  -d '{
    "notebook_id": "nb_xyz789",
    "since_position": 120
  }'
```

This returns all entries added since position 120, allowing you to process changes programmatically.

Method 3: View Activity Timeline In any notebook, click **"Activity"** or **"Timeline"**:

```
Q1 Planning - Recent Activity


Position 130 - Jan 22, 2:30 PM
  Jane Smith revised "Engineering Roadmap Q1"
  Reason: Added database optimization initiative

Position 129 - Jan 22, 1:15 PM
  Bob Johnson created "Q1 Budget Summary"
  References: 2 entries

Position 128 - Jan 22, 10:00 AM
  Alice Chen revised "Team Onboarding Guide"
  Reason: Updated with new team members

Position 127 - Jan 21, 4:45 PM
  Jane Smith revised "Engineering Roadmap Q1"
  Reason: Updated timeline after planning meeting
```

Key insights: - **Position** = causal order (not timestamps) - **Chronological view** of what changed - **Types of changes** visible at a glance - **Who changed what** for audit purposes

Verification

Confirm you're observing correctly:

☐ You're receiving notifications or can view activity timeline
☐ You can see new entries as they're added
☐ You can see revisions with reasons
☐ Activity is in causal order (positions increase)
☐ You understand the impact of changes

Tips & Tricks

Set Smart Notification Frequency

- **Daily digest** — Good for active notebooks you check regularly
- **Weekly summary** — Good for passive monitoring
- **Immediately** — Only for critical entries (security, compliance)

Track Specific Topics Some notebooks let you "watch" specific topics:

```
Watch Topics in Q1 Planning


  organization/planning/goals
  organization/planning/budget
  organization/planning/hiring

Notify when entries in these topics are created or revised.
```

Use Positions for Bookmarking Note the **position number** of where you last caught up:

```
Last checked: Position 120
Today's new entries: Position 121-130
```

Next time you check, start from position 120 to see only new changes.

Next Steps

After observing changes: - Revise your entries if new information affects them - Discuss contradictions with other contributors - Update dependent work if requirements changed

---

Summary: Quick Reference

The 5 Workflows at a Glance

Workflow	Purpose	Time	Frequency
1. Create Entries	Add new knowledge	10-30 min	Weekly
2. Browse & Search	Discover existing knowledge	5-15 min	Daily
3. Search Notebooks	Cross-team knowledge discovery	5-10 min	As needed
4. Manage Revisions	Update entries over time	10-20 min	As needed
5. Observe Changes	Stay informed of updates	2-5 min	Continuous

Your Workflow Loop

```
1. Create Entry
   ↓ (Research needed)
2. Browse & Search
   ↓ (Found related entries)
3. Create Revision
   ↓ (Or create new entry building on discovery)
4. Observe Changes
   ↓ (Track impact and discussions)
5. Back to Step 1
   ↓ (Continuous knowledge refinement)
```

Keyboard Shortcuts

Shortcut	Action
`/`	Search
`n`	New entry
`e`	Edit/revise entry
`s`	Save
`Esc`	Close modal
`?`	Show all shortcuts

Related Personas

Your workflows often overlap with:

- **Notebook Owner** — Who reviews your submissions and manages access
- **Auditor/Compliance Officer** — Who reviews your entries for security/compliance
- **Cross-Org Coordinator** — Who may mirror your entries to other organizations

---

Troubleshooting

"Access Denied" When Creating Entry

Cause: You don't have "Read+Write" access to this notebook.

Solution: 1. Ask the notebook owner to grant you write access 2. Check your clearance level (Settings → Profile) 3. Ensure you're trying to write to the right notebook

Entry Stuck in "Probation" Status

Cause: Background analysis is taking longer than usual.

Solution: 1. Wait 5-15 minutes, then refresh 2. Check system status dashboard 3. Contact admin if stuck for > 1 hour

Revision Didn’t Save

Cause: Network error or session timeout.

Solution: 1. Try again; draft may be auto-saved locally 2. Copy your content to clipboard before retrying 3. Check your internet connection

Can’t Find an Entry I Know Exists

Cause: Search index lag or access restriction.

Solution: 1. Try browsing by topic instead of searching 2. Check your clearance level (you may not have access) 3. Ask notebook owner to confirm entry exists

---

Last updated: February 21, 2026 **Chapter version:** 1.0.0 (Beta) **Platform version:** 2.1.0

- Design and maintain organizational hierarchy (DAG structure)
- Grant and revoke security clearances (levels + compartments)
- Manage group membership and roles
- Configure ThinkerAgents and security parameters
- Ensure Bell-LaPadula compliance
- Monitor organizational audit trails

Required Permissions: - “Admin” role in your organization - Top-secret or SECRET clearance (minimum) - Understanding of security model fundamentals

Typical Workflows: 4 core workflows in this chapter

Workflow 1: Creating Organizational Structure

Overview

Design your organization’s group hierarchy—who reports to whom, which teams collaborate, and how clearances flow through the organization.

Use case: You’re setting up Cyber for a new organization or restructuring an existing team hierarchy.

Related workflows: - Managing Group Memberships — Add users to groups after structure exists - Managing Clearances — Assign clearances that respect the hierarchy

Prerequisites

☐ Organization created (by system admin)
☐ Organization admin access
☐ Clear understanding of your org structure
☐ List of teams and reporting relationships

Step-by-Step Instructions

Step 1: Access Organization Administration **Navigate to:** Admin panel → Organizations → Select your org

1. Click the **Admin Panel** icon (gear) in top-right
2. Select **Organizations** from sidebar
3. Click your organization’s name
4. You’ll see the **Organization Dashboard**:

```
MyCompany Organization


[Overview] [Groups] [Members] [Audit Log] [Settings]

Group Hierarchy:
```

```
MyCompany (root)
   Engineering
      Backend
      Infrastructure
   Operations
   Finance
```

Step 2: Click "Groups" Tab See the groups view where you manage structure:

```
Groups in MyCompany


[+ New Group]

Group Hierarchy (DAG - Directed Acyclic Graph):

Name                 | Members | Notebooks | Actions

MyCompany (root)     | 45      | 3         | [Edit]
  Engineering       | 12      | 8         | [Edit]
    Backend        | 5       | 4         | [Edit]
    Infrastructure | 7       | 4         | [Edit]
  Operations        | 15      | 5         | [Edit]
  Finance           | 3       | 2         | [Edit]
```

Step 3: Create a New Group Click "[+ **New Group**]":

```
Create New Group


Group Name *
[Engineering]

Description
[Engineering teams: backend, infrastructure, security]

Parent Group(s) *
[Dropdown: Select parent(s)...]
   MyCompany (root)
   Operations
   (other options)

Classification Level (inherited from parents)
[Read-only: CONFIDENTIAL] ← Automatically set to highest
                            parent's level

Compartments (inherited from parents)
[Read-only: {Strategic Planning, Operations}]

[Create Group]  [Cancel]
```

Key Concepts:

Term	Meaning
Parent Group	The group above in hierarchy (can have multiple)
DAG	Directed Acyclic Graph — complex relationships allowed, but no cycles
Classification Inheritance	Child inherits the highest classification of any parent
Compartment Inheritance	Child gets union of all parent compartments

Step 4: Set Classification & Compartments Classification and compartments are **inherited** from parents and automatically elevated:

```
Example:
Parent "Engineering" = SECRET / {Operations}
Parent "Backend" = CONFIDENTIAL / {Operations, Infrastructure}

New child of both:
Inherits: SECRET / {Operations, Infrastructure}
(highest level + union of compartments)
```

You can **add more compartments** to a child group beyond what's inherited:

```
Group: Backend Team


Inherited:    SECRET / {Operations, Infrastructure}
Add Compartment: [+ Add]
    Operations (inherited)
    Infrastructure (inherited)
    Database Access (new)
    Cryptography (not needed)

Final Classification: SECRET / {Operations, Infrastructure, Database Access}
```

Step 5: Verify Hierarchy (DAG) After creating multiple groups, verify the hierarchy:

```
MyCompany Organization Structure


      MyCompany
      (CONFIDENTIAL / {Strategic})
      /                    \
Engineering               Operations
(SECRET / {Strategic,    (CONFIDENTIAL / {Operations})
 Operations})                 |
  /       \
Backend   Infrastructure  Incident    Admin
(SECRET/  (SECRET / {Ops, Response   (CONF / {Ops})
 {Strat,   Infra, DB})   {Ops})
 Ops,
 Infra,
 DB})
```

Verify: - No cycles (Backend → Engineering → MyCompany → no cycle back) - Classification increases or stays same going down - Compartments accumulate as you go down

Step 6: Update Group (If Needed) To modify an existing group:

1. Click **"[Edit]"** next to the group name
2. You can change:
    - Description
    - Parent relationships (add/remove parents)
    - Additional compartments
3. Click **"Save Changes"**

What you can't change: - Group name (would break references) - Remove parents (would break hierarchy) - Reduce classification level (security violation)

Verification

Confirm your structure is sound:

☐ All teams have parent groups
☐ No cycles exist (use the visualization)

☐ Classification increases or stays same going down
☐ Compartments accumulate correctly
☐ Root group exists and everyone can trace lineage to it
☐ Notebook owners understand their group’s classification

Tips & Tricks

Design Pattern: Functional + Geographic Mix functional and geographic hierarchies:

```
Organization
   By Function
      Engineering
      Operations
      Finance
   By Location
      North America
      Europe
   By Security Domain
       Public Facing
       Internal
       Confidential
```

A user can be in multiple groups (DAG allows this), so one engineer can be in: - Engineering / Backend - North America / Operations - Internal / Security Domain

Classification Best Practices Start conservative:

```
 Start with everything TOP_SECRET
 Start with CONFIDENTIAL
   Elevate groups only as needed
```

Compartment Naming Use clear, consistent names:

```
 Good names:
   - Medical Research
   - Infrastructure Operations
   - Customer PII
   - Executive Confidential

 Bad names:
   - Top Secret Stuff
   - Internal
   - Secret1, Secret2
   - TBD
```

Next Steps

After creating structure: - Manage group memberships to add users - Assign clearances at appropriate levels - Create notebooks for teams (described in Chapter 6)

Workflow 2: Managing Group Memberships

Overview

Add users to groups and assign roles within those groups (member vs. admin).

Use case: A new engineer joins your team; you add them to the Engineering group.

Related workflows: - Creating Organizational Structure — Groups must exist first - Managing Clearances — Clearances are independent of group membership

Prerequisites

- □ Group exists (from Workflow 1)
- □ Users have been created in the system
- □ Organization admin access

Step-by-Step Instructions

Step 1: Go to Group Management **Navigate to:** Admin → Organizations → Groups → Select group

1. Click **Admin** in top-right
2. Go to **Organizations** → Your org → **Groups** tab
3. Click the group name
4. You'll see the group's member list:

```
Engineering Group


Members (5):

Name          | Email               | Role    | Actions

Alice Chen    | alice@myco.com     | Admin    | [Remove]
Bob Johnson   | bob@myco.com       | Member   | [Edit Role]
Carol Davis   | carol@myco.com     | Member   | [Edit Role]
...
```

Step 2: Add a User Click "[+ **Add Member**]":

```
Add Member to Engineering


Search for user:
[Dropdown: Start typing name/email...]

Results:
  David Smith (david@myco.com)
  Eve Wilson (eve@myco.com)
  Frank Brown (frank@myco.com)

Assign Role:
  Member (can use group resources, can't manage)
  Admin   (can manage group, add/remove members)

[Add] [Cancel]
```

Role Explanations:

Role	Can Do	Can't Do
Member	Use notebooks owned by group, create entries	Add/remove members, manage group settings
Admin	Everything + add/remove members, change roles	Delete group, modify classification

Step 3: Bulk Add Members (Advanced) For adding multiple people:

1. Click "[**Import Members**]"
2. Paste a list:

```
alice@myco.com, admin
bob@myco.com, member
carol@myco.com, member
david@myco.com, admin
```

3. Review mappings
4. Click **"[Confirm Import]"**

Step 4: Edit Member Roles If someone's role needs to change:

1. Click **"[Edit Role]"** next to their name
2. Select new role:

   ```
   Change Role for Bob Johnson

   Current: Member
   New:       Member
              Admin
   [Save] [Cancel]
   ```

3. Click **"[Save]"**

Step 5: Remove a Member Click **"[Remove]"** next to their name:

```
   Remove Alice Chen from Engineering?

This will:
  • Remove her access to Engineering-owned notebooks
  • Revoke her group admin rights (if applicable)
  • NOT delete her account or other group memberships

[Confirm Remove] [Cancel]
```

Click **"[Confirm Remove]"**.

Verification

Confirm membership is correct:

☐ User appears in group member list
☐ User has correct role (Member or Admin)
☐ User can access group-owned notebooks
☐ User can't perform actions above their role
☐ Removal revoked access to group resources

Tips & Tricks

Nested Admin Roles Group admins can manage their own group but not parent/sibling groups:

```
Structure:
MyCompany (Org Admin)
   Engineering (Group Admin: Alice)
       Backend (Group Admin: Bob)
       Infrastructure (Group Admin: Carol)
   Operations (Group Admin: David)

Permissions:
- Alice (Engineering Admin): Can manage Engineering + Backend + Infrastructure
- Bob (Backend Admin): Can manage only Backend
- Org Admin: Can manage everything
```

Audit Group Changes All membership changes are logged. Check the group's audit trail:

Click **"[Audit Log]"** in the group settings.

Cascade Effects When adding a user to a group, they automatically get: - Access to all notebooks owned by that group - Clearance requirements of that group (or higher) - Audit trail visibility for that group

Next Steps

After managing memberships: - Assign appropriate clearances (Workflow 3) - Create notebooks for the group (Chapter 6) - Monitor group activity in audit logs

---

Workflow 3: Managing Security Clearances

Overview

Grant security clearances to principals (users or groups) specifying classification levels and compartments they can access.

Use case: A new contractor needs access to your infrastructure documentation, but only the non-sensitive parts. You grant them CONFIDENTIAL clearance without infrastructure compartments.

Related workflows: - Organizational Structure — Clearances work within your org structure - Group Membership — Group membership affects clearance inheritance

Prerequisites

- ☐ Group membership established
- ☐ User or group needs clearance assignment
- ☐ Organization admin access
- ☐ Understanding of Bell-LaPadula model (Chapter 2)

Step-by-Step Instructions

Step 1: Access Clearance Management **Navigate to:** Admin → Organizations → Members (or Groups)

1. Click **Admin** in top-right
2. Go to **Organizations** → Your org → **Members** tab
3. Find the user you want to grant clearance to:

```
Users in MyCompany


[+ Invite User]

Name          | Email             | Clearance         | Actions

Alice Chen    | alice@myco.com   | SECRET / {Ops}    | [Edit]
Bob Johnson   | bob@myco.com     | CONFIDENTIAL / {} | [Edit]
Carol Davis   | carol@myco.com   | (no clearance)    | [Assign]
```

Click “**[Assign]**” or “**[Edit]**” next to the user.

Step 2: Set Classification Level

```
Set Clearance for Carol Davis


Classification Level *
[Dropdown: Select level...]

  PUBLIC       (no access restrictions)
  CONFIDENTIAL (internal use only)
  SECRET       (restricted distribution)
  TOP_SECRET   (severe impact if disclosed)

Current Group Clearance: SECRET / {Operations}
(Your clearance must be  what you grant)
```

Important Rule: You can only grant clearance **up to your own level**. If you’re CONFIDENTIAL, you can’t grant SECRET.

Step 3: Select Compartments Check the compartments the user needs:

```
Compartments


  Operations       (required for group membership)
  Database Access  (required for group membership)
  Executive Only   (additional, not required)
  Medical Records  (additional)
  Cryptography     (additional)

Final Clearance: CONFIDENTIAL / {Operations, Database Access}

Note: User's group requires  , you can add more
```

Rules: - User must have parent group's compartments - You can add additional compartments - You can't remove required compartments (from group) - You can't grant compartments you don't have

Step 4: Apply Clearance Click "[**Apply Clearance**]":

```
  Clearance assigned!

Carol Davis now has: CONFIDENTIAL / {Operations, Database Access}

She can access:
  • All PUBLIC notebooks
  • All CONFIDENTIAL notebooks
  • CONFIDENTIAL entries with Operations or Database Access labels
  • NOT: SECRET or TOP_SECRET entries

Changes take effect immediately.
[OK]
```

Step 5: Update Clearance (When Needed) If circumstances change (promotion, role change):

1. Click "[**Edit**]" next to the user
2. Modify level and/or compartments
3. Add reason for change:

   ```
   Clearance Change Reason:
   [Promoted to senior engineer, needs cryptography access]
   ```

4. Click "[**Update Clearance**]"

Step 6: Revoke Clearance (If Needed) To revoke:

1. Click "[**Edit**]" next to the user
2. Click "[**Remove Clearance**]"
3. Confirm:

   ```
        Remove clearance for Carol Davis?

   She will:
     • Lose access to all classified notebooks
     • Keep PUBLIC notebook access
     • Still be in all groups (group membership unchanged)

   [Confirm] [Cancel]
   ```

Verification

Confirm clearance is working:

- □ User has specified clearance level
- □ All required compartments are present
- □ User can access appropriate notebooks
- □ User can't access more restricted content
- □ Audit log shows clearance change
- □ User receives notification of clearance change

Clearance Examples

Example 1: New Team Member

```
New Engineer (Alice)
  Assigned to: Engineering / Backend group
  Group requires: SECRET / {Operations}

Clearance to grant:
  Level:       SECRET (minimum: can't be less than group)
  Compartments: {Operations} (minimum: can't be less than group)

Full clearance: SECRET / {Operations}

Can read:
    PUBLIC anything
    CONFIDENTIAL anything
    SECRET / {Operations}
    SECRET / {Infrastructure, Cryptography}
    TOP_SECRET anything
```

Example 2: Contractor with Limited Access

```
Contractor (Bob)
  Short-term engagement
  Only needs documentation

Clearance to grant:
  Level:       CONFIDENTIAL (limited exposure)
  Compartments: {} (no sensitive compartments)

Full clearance: CONFIDENTIAL / {}

Can read:
    PUBLIC anything
    CONFIDENTIAL anything
    SECRET anything
    TOP_SECRET anything
```

Example 3: Cross-Functional Manager

```
Manager (Carol)
  Oversees Engineering AND Operations
  In both groups:
    - Engineering: SECRET / {Operations}
    - Operations: SECRET / {Operations}

Clearance to grant:
  Level:       SECRET (matches groups)
  Compartments: {Operations} (union of group requirements)

Can grant additional:
    Facilities Management (new compartment)

Full clearance: SECRET / {Operations, Facilities Management}
```

Tips & Tricks

Clearance Cache Changes take effect **immediately** in most cases, but access control caches may take up to **5 minutes** to refresh. To force immediate refresh:

Admin → Organizations → **Flush Clearance Cache**

Audit Clearance Changes Track who changed what:

```
[View Clearance Audit Log]

Carol Davis Clearance History:


Jan 22, 2:30 PM - Updated by Alice Chen
  From: CONFIDENTIAL / {Operations}
  To:   CONFIDENTIAL / {Operations, Database}
  Reason: Promoted to senior engineer

Jan 15, 10:00 AM - Assigned by Admin
  Level: CONFIDENTIAL / {Operations}
  Reason: New team member
```

Principle of Least Privilege Always apply minimum necessary clearance:

- Engineer working on database: Grant DATABASE compartment
- Engineer working on database: Grant all compartments
- New hire: Start with CONFIDENTIAL, promote as needed
- New hire: Give them SECRET "just in case"

Next Steps

After assigning clearances: - Create notebooks respecting the clearance hierarchy - Test that access control works as expected - Review clearances quarterly

---

Workflow 4: Configuring ThinkerAgents

Overview

Register AI processing workers (ThinkerAgents) with your organization, specifying their security classification and capabilities.

Use case: You have a background worker that processes notebook entries for embeddings and wants to register it with Cyber.

Related workflows: - Organizational Structure — Agents inherit org structure - System Administrator — Platform-wide agent management

Prerequisites

☐ Agent software is ready to deploy
☐ Organization admin access
☐ Understanding of agent's capabilities and security needs
☐ Network/infrastructure details for agent

Step-by-Step Instructions

Step 1: Access Agent Management **Navigate to:** Admin → Organizations → Agents (or Admin → Agents)

```
MyCompany Agents


[+ Register Agent]
```

```
Name                    | Type        | Classification | Status

embedding-worker-1      | Embeddings  | CONFIDENTIAL   | Active
claims-distiller        | Claims      | SECRET         | Active
comparison-engine       | Comparison  | CONFIDENTIAL   | Idle
(none yet)
```

Click “[+ **Register Agent**]”.

Step 2: Enter Agent Details

```
Register New ThinkerAgent


Agent Name *
[embedding-processor]

Agent Type *
[Dropdown: Select type...]
  • Embedding     (creates vector embeddings)
  • Claims       (extracts/distills claims)
  • Comparison   (compares entry semantics)
  • Custom       (other processing)

Description
[Processes all notebook entries to create embeddings for similarity search]

Infrastructure Location
[us-east-1-prod]
```

Agent Types:

Type	Purpose	Classification
Embedding	Create vector embeddings for search/similarity	Usually CONFIDENTIAL
Claims	Extract and distill claims from entries	Usually SECRET
Comparison	Analyze semantic similarity between entries	Usually SECRET
Custom	Organization-specific processing	As needed

Step 3: Set Security Classification

```
Security Classification


Max Classification Level *
[Dropdown: Select level...]

  CONFIDENTIAL (can process up to CONFIDENTIAL entries)
  SECRET       (can process up to SECRET entries)
  TOP_SECRET   (can process TOP_SECRET entries)

Compartments *
[Multi-select: Choose compartments...]

  Operations
  Database
  Executive
  Medical
```

```
Note: Agent can process entries with any subset of these compartments.
```

Important: Agent's classification can't exceed your organization's clearance for agents. If your org is CONFIDENTIAL, agents can't be higher than that.

Step 4: Configure Capabilities Specify what the agent can do:

```
Capabilities


Access Control:
  READ entries (can read/process content)
  WRITE results (can store results/outputs)
  REVISE entries (can update entries)
  DELETE (not recommended for processing agents)

Job Types:
  DISTILL_CLAIMS (extract claims from entries)
  COMPARE_CLAIMS (compare entry claims)
  EMBED_ENTRIES (create embeddings)
  CUSTOM_JOB_TYPE (define custom)

Rate Limits:
Entries per minute: [100]
Concurrent jobs: [5]
```

Step 5: Provide Credentials The system generates credentials for the agent:

```
Agent Credentials


Agent ID:
embedding-worker-1-abc123

API Token:
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...

    Save these credentials securely!
    They won't be shown again.
    Use in agent deployment:
      CYBER_AGENT_ID=embedding-worker-1-abc123
      CYBER_AGENT_TOKEN=eyJ...
```

Click **"[Copy Credentials]"** and securely save them.

Step 6: Configure Infrastructure Specify where the agent runs:

```
Infrastructure Details


Deployment Location:
[us-east-1-production]

Endpoint URL:
[https://agent-worker-1.mycompany.internal:8080/health]

Health Check Interval:
[Every 5 minutes]

Failover Strategy:
  Stop on error (don't retry)
  Retry with backoff
  Use backup agent
```

Step 7: Register Agent Click “**[Register Agent]**”:

```
  Agent registered!

embedding-worker-1

ID: embedding-worker-1-abc123
Status: Pending (waiting for first heartbeat)
Next check: In 5 minutes

Next steps:
1. Deploy agent with credentials
2. Agent connects to Cyber
3. Status becomes "Active"
4. Jobs will be sent to agent

[View Agent Status] [Back]
```

Step 8: Verify Agent Connection After deployment, monitor the agent’s status:

```
Agent Status Dashboard


Agent: embedding-worker-1-abc123
Status:   Active (last seen: 2 minutes ago)
Uptime: 12 hours
Processed: 2,341 entries
Failed jobs: 0
Current load: 3/5 concurrent jobs
```

Verification

Confirm the agent is working:

- □ Agent appears in agent list with “Active” status
- □ Agent credential sare securely stored
- □ Health check passing (status)
- □ Jobs are being assigned to agent
- □ Failed jobs are logged and visible
- □ Agent respects security classification limits

Agent Security Considerations

Important Rules

1. **Agents cannot exceed organizational classification**
   - If your org max is CONFIDENTIAL, agents can’t be SECRET
2. **Agents inherit organizational structure**
   - Agent processes entries from groups in that organization
   - Subject to Bell-LaPadula rules
3. **Agent actions are audited**
   - Every job processed is logged
   - Changes made by agent are signed with agent identity
4. **Credentials must be kept secure**
   - Like API tokens, treat as passwords
   - Store in secure environment variables
   - Rotate yearly

Limiting Agent Scope To limit what an agent can process:

- Restrict **compartments** — Agent only sees entries in allowed compartments
- Limit **job types** — Agent only does specific work (e.g., embedding, not revisions)

- Set **rate limits** — Prevent resource exhaustion
- Remove **WRITE capability** — Agent can read but not create/modify

Tips & Tricks

Monitor Agent Health Check agent status regularly:

```
[Agent Health Check]

Last heartbeat:  2 minutes ago
Response time: < 100ms
CPU usage: 45%
Memory: 2.1 GB / 4 GB
Errors (last hour): 0
```

Rotate Agent Credentials Yearly rotation recommended:

```
[Rotate Credentials]

Current token expires: Jan 2027
Generate new token: [Generate]
Revoke old token: [Revoke after verification]
```

Multi-Agent Redundancy Deploy multiple agents for fault tolerance:

```
embedding-worker-1 (us-east-1)   Active
embedding-worker-2 (us-west-1)   Active
claims-distiller-1 (us-east-1)   Active
claims-distiller-2 (us-west-1)   Backup
```

Jobs distribute across active agents.

Next Steps

After registering agents: - Deploy agent software to specified infrastructure - Monitor agent health dashboard - Create notebooks that agents process - Review agent job logs in audit trail

---

Summary: Quick Reference

The 4 Workflows at a Glance

Workflow	Purpose	Time	Frequency
1. Org Structure	Design group hierarchy	30-60 min	Setup only
2. Group Membership	Add users to groups	5-10 min	As needed
3. Clearances	Grant security access	5-15 min	As needed
4. ThinkerAgents	Register workers	20-30 min	Quarterly

Your Workflow Loop

```
1. Design Org Structure (once)
   ↓
2. Add Group Members (ongoing)
   ↓
3. Grant Clearances (ongoing)
   ↓
4. Register Agents (quarterly)
   ↓
5. Monitor & Update (continuous)
```

Key Principles

- **Hierarchy First:** Structure before membership
- **Least Privilege:** Grant minimum necessary clearance
- **Audit Everything:** All changes are logged
- **Security by Default:** Classify conservatively
- **Inherit Down:** Classification & compartments flow down hierarchy

---

Related Personas

Your workflows overlap with:

- **System Administrator** — Platform-wide user and agent management
- **Notebook Owner** — Who manage notebooks within your org structure
- **Knowledge Contributor** — Who use the groups and clearances you set up

---

Troubleshooting

"Can't Grant This Clearance"

Cause: You're trying to grant clearance higher than your own, or compartments you don't have.

Solution: 1. Check your own clearance level (Settings → Profile) 2. Request higher clearance from your organization's security officer 3. Or grant only what you have clearance for

User Can't Access Notebook

Cause: User is in group, but clearance doesn't match notebook's classification.

Solution: 1. Check notebook's classification (Notebook Settings) 2. Check user's clearance (Admin → Members) 3. Elevate user's clearance or user's group clearance 4. Flush clearance cache (Admin → Organizations → Flush Cache)

Agent Shows "Inactive"

Cause: Agent hasn't connected yet, or network issue.

Solution: 1. Verify agent credentials in deployment 2. Check agent logs for connection errors 3. Verify network allows agent → Cyber connection 4. Check agent's classification level (may be too high)

Hierarchy Creates Cycle

Cause: DAG structure is broken (not actually a DAG).

Solution: 1. Review group relationships carefully 2. Use the hierarchy visualizer 3. Remove or adjust parent relationships to break cycle 4. Consult the Bell-LaPadula model (Chapter 2)

Last updated: February 21, 2026 **Chapter version:** 1.0.0 (Beta) **Platform version:** 2.1.0

- Create notebooks with appropriate classification
- Manage access control (who can read, write, admin)
- Review and approve external contributions (if gating is enabled)
- Monitor job processing (embeddings, claims, analysis)
- Manage subscriptions to other notebooks
- Maintain notebook quality and organization

Required Permissions: - "Admin" access to at least one notebook (usually yours) - Read+Write access to create entries - Your organization's clearance

Typical Workflows: 5 core workflows in this chapter

Workflow 1: Creating and Configuring Notebooks

Overview

Create a new notebook and configure its classification, ownership, and basic settings.

Use case: Your team needs a shared knowledge space for documenting architectural decisions. You create a notebook with appropriate security labels.

Related workflows: - Managing Access Control — Grant access after creation - Reviewing Submissions — Set up content review if needed

Prerequisites

- ☐ Cyber account with Read+Write access
- ☐ Understand your team's classification level
- ☐ Clear purpose for the notebook
- ☐ Owner group identified

Step-by-Step Instructions

Step 1: Go to Notebooks **Navigate to:** Sidebar → Notebooks → [+ New Notebook]

1. Click **Notebooks** in the left sidebar
2. You'll see your existing notebooks
3. Click **"[+ New Notebook]"** button

Step 2: Fill in Notebook Details

```
Create New Notebook


Name *
[Architectural Decisions]

Description
[Central repository for architecture decisions, ADRs, and design documents]

Owner Group *
[Dropdown: Select group...]
   Engineering
   Infrastructure
   Architecture Council

Classification Level (Advanced)
[Dropdown: CONFIDENTIAL] ← Usually inherits from group

Compartments (Optional)
[Tag input: Add compartment names...]
  Examples: Strategic, Infrastructure, etc.

Retention Policy
[Dropdown: Select retention...]
   1 year
   3 years (default)
   7 years
   Permanent

[Create Notebook] [Preview] [Cancel]
```

Field Explanations:

Field	Required	Notes
Name	Yes	Concise, clear (e.g., “API Architecture”, not “Stuff”)
Description	No	1-2 sentences explaining purpose
Owner Group	Yes	The team that owns this notebook
Classification	No	Inherited from group; can be more restrictive
Compartments	No	Additional security categories
Retention	No	How long entries are kept before deletion

Step 3: Set Classification (If Advanced) Classification usually inherits from the owner group:

```
Owner Group: Engineering / Backend
Group Classification: SECRET / {Operations}

Notebook Options:
  • Inherit: SECRET / {Operations} ← (automatically set)
  • More Restrictive: SECRET / {Operations, Database}
  • NOT ALLOWED: CONFIDENTIAL (lower than group)
```

You can **add compartments** but not remove or lower the level.

Step 4: Create Notebook Click **“[Create Notebook]”**:

```
 Notebook created!

Name: Architectural Decisions
Owner: Engineering / Backend
Classification: SECRET / {Operations}
Access: You have Admin access

Next steps:
  1. Invite collaborators [Manage Access]
  2. Create first entry [Start Writing]
  3. Configure settings [Notebook Settings]

[View Notebook] [Back]
```

Step 5: Configure Settings (Optional) Go to the notebook and click **Settings** tab:

```
Notebook Settings


Notebook Name: Architectural Decisions
Owner Group: Engineering / Backend
Classification: SECRET / {Operations}

Retention Policy: 3 years (entries older than 3 years are archived)

Ingestion Gating:
 Require review for new entries (optional content review gate)

Notifications:
 Notify on new entries
 Notify on revisions
 Notify on comments

[Save Changes]
```

Ingestion Gating: If enabled, all new entries go to a review queue before being published.

Verification

Confirm your notebook is set up:

- ☐ Notebook appears in your Notebooks list
- ☐ You have "Admin" access
- ☐ Name and description are correct
- ☐ Classification is appropriate for content
- ☐ Retention policy is set
- ☐ You can create an entry in it

Tips & Tricks

Naming Conventions Use consistent naming across your organization:

```
Good names:
  - Team name first: "Backend / Database Queries"
  - Clear scope: "Q1 Planning"
  - Single purpose: "Security Incident Log"

Bad names:
  - Vague: "Stuff", "Notes", "Temporary"
  - Redundant: "Backend Backend Things"
  - Too broad: "Everything"
```

Description Best Practices Write descriptions that help people decide if they should read:

```
Good:
  "Central repository for architecture decisions (ADRs) and design
   documents for the backend team. Covers database design, API specs,
   and infrastructure patterns."

Bad:
  "Architecture"
  "Important stuff"
  "Read this"
```

Classification Strategy Start conservative:

```
Team Classification:    SECRET / {Operations}
Notebook Options:

  Make it PUBLIC for accessibility
  Start with SECRET / {Operations}
    Restrict further only if needed
  Document why it's classified that way
```

Next Steps

- Manage Access Control — Add collaborators
- Review Submissions — Set up review gates
- Start creating entries

Workflow 2: Managing Access Control

Overview

Grant and revoke access to your notebook for users and groups at four tiers: Existence, Read, Read+Write, Admin.

Use case: Your Architecture Council notebook should allow executives to read but not edit. You grant them "Read" access.

Related workflows: - Creating Notebooks — Access set after creation - Reviewing Submissions — Admin access needed

Prerequisites

- ☐ Notebook already created
- ☐ Admin access to the notebook
- ☐ Know who needs access and at what level

Step-by-Step Instructions

Step 1: Go to Access Control Tab **Navigate to:** Notebooks → Select notebook → Access Control tab

```
Architectural Decisions


[Entry Feed] [Settings] [Access Control] [Statistics]

Current Access List:


Principal         | Type  | Tier       | Actions

Engineering Team  | Group | Read+Write | [Edit] [Remove]
You (Jane Smith)  | User  | Admin      | (you)

[+ Add User or Group]
```

Step 2: Click "Add User or Group"

```
Grant Access


Search for principal:
[Type to search...]

Results:
  Alice Chen (user)
  Bob Johnson (user)
  Executive Team (group)
  Security Council (group)

Access Tier:
  Existence    (know it exists, but can't read)
  Read         (can read, can't write)
  Read+Write   (can read and create/revise)
  Admin        (full control)

[Grant Access] [Cancel]
```

Access Tiers:

Tier	Can Read	Can Write	Can Manage	Use Case
Existence				Secret/unlisted notebooks
Read				Stakeholders, viewers
Read+Write				Contributors
Admin				Notebook owner, managers

Step 3: Grant Access

1. Check the principal you want to grant access to
2. Select the appropriate tier
3. Click **"[Grant Access]"**

```
 Access granted!

Executive Team: Read access to Architectural Decisions

They can:
  • Read all entries (including restricted ones, if they have clearance)
  • See history and revisions
    Create new entries
    Manage access

[OK]
```

Step 4: Edit Access Levels If you need to change someone's access:

1. Click **"[Edit]"** next to their name
2. Select new tier
3. Provide reason (optional):

   ```
   Reason for changing access:
   [Promoted to tech lead, needs write access]
   ```

4. Click **"[Save]"**

Step 5: Revoke Access To remove someone's access:

1. Click **"[Remove]"** next to their name
2. Confirm:

   ```
     Remove Alice Chen's Read+Write access?

   She will:
     • Lose ability to read this notebook
     • Keep access through group membership (if any)
     • Audit log will record the removal

   [Confirm] [Cancel]
   ```

3. Click **"[Confirm]"**

Access Control Scenarios

Scenario 1: Internal Team Notebook

```
Notebook: Backend Engineering Decisions
Owner: Backend Team (SECRET / {Operations})

Access Control:


Backend Team      | Group | Read+Write (auto via group)
Infrastructure    | Group | Read       (needs visibility)
You (Owner)       | User  | Admin      (owner)
Security Lead     | User  | Read       (compliance review)

Result:
  • 5 backend engineers: full access
  • 3 infrastructure engineers: can learn from decisions
  • 1 security lead: can audit compliance
  • Others: no access
```

Scenario 2: Cross-Functional Documentation

```
Notebook: API Architecture Spec
Owner: Backend Team (SECRET / {Operations})

Access Control:


Backend Team         | Group | Read+Write
Frontend Team        | Group | Read
Mobile Team          | Group | Read
Product Team         | Group | Read
Client Partnerships | Group | Existence (they know it exists)

Result:
```

- Backend engineers: can update spec
- Frontend/Mobile engineers: know about API
- Product: understands what's possible
- Client Partnerships: knows to reference it privately

Scenario 3: Executive Dashboard

```
Notebook: Quarterly Roadmap
Owner: Leadership Team (SECRET / {Operations, Strategic})

Access Control:


Leadership Team        | Group | Read+Write (collaborators)
Engineering Director  | User  | Admin      (co-owner)
Product Director      | User  | Admin      (co-owner)
Department Heads      | Group | Read       (visibility)
All Staff             | Group | Existence  (know it exists)

Result:
```

- 3 people can write/edit roadmap
- Department heads can read to understand direction
- Everyone else knows it exists but can't read

Verification

Confirm access is correct:

- □ Each principal has appropriate tier
- □ Owner still has Admin access
- □ Contributors have Read+Write, not Admin
- □ Viewers have Read, not Write
- □ Audit log shows access changes
- □ Removed principals can no longer access

Tips & Tricks

Principle of Least Privilege Only grant necessary access:

```
"Give everyone Read+Write to be collaborative"
"Give contributors Read+Write, others Read"

"Make everyone Admin so they can help manage"
"Keep Admin to just notebook owners"

"Restrict everyone to Existence (too secretive)"
"Allow appropriate tiers based on role"
```

Group vs. Individual Access Prefer groups:

```
  Grant access to "Backend Team" group
    • Automatically includes new team members
    • Easy to update one place

  Grant access to individual engineers
    • Need to manually add/remove each person
    • Easy to miss people
```

Track Access Changes Monitor who has what:

```
[View Access History]

Access Control Audit Trail


Jan 22, 2:30 PM - Jane Smith granted "Alice Chen" Read access
Jan 20, 10:00 AM - Admin revoked "Carol Davis" Read+Write (left team)
Jan 15, 9:00 AM - Jane Smith created notebook, auto-granted "Backend Team" Read+Write
```

Cascade Access from Groups If someone is in the owner group, they automatically get that access:

```
Backend Team = Read+Write

Alice Chen is in "Backend Team" group
→ Automatically has Read+Write access
→ Can't remove individual access (must remove from group)
```

Next Steps

After setting access control: - Invite people to start contributing - Create first entry - Set up review gates if needed

---

Workflow 3: Reviewing Submissions

Overview

If ingestion gating is enabled, review and approve/reject new entries before they're published to the notebook.

Use case: Your Architecture Council wants to ensure entries meet quality standards before publication.

Related workflows: - Creating Notebooks — Enable gating during setup - Creating Entries — The submission side

Prerequisites

- ☐ Ingestion gating enabled on notebook (Workflow 1)
- ☐ Admin access to the notebook
- ☐ Understanding of what makes a good submission

Step-by-Step Instructions

Step 1: Access Review Queue **Navigate to:** Notebooks → Select notebook → [Review] tab

```
Architectural Decisions


[Entry Feed] [Review] [Settings] [Access Control]

Pending Submissions (3):


[ ] Database Indexing Strategy
```

```
    Submitted by: Carol Davis
    Submitted: Jan 22, 10:30 AM
    Status: Pending
    [View] [Approve] [Request Changes] [Reject]

[ ] Caching Architecture
    Submitted by: Bob Johnson
    Submitted: Jan 22, 9:15 AM
    Status: Waiting for changes
    Last feedback: Jan 22, 10:00 AM (from Jane Smith)
    [View] [Approve] [Request Changes] [Reject]

[ ] API Versioning Policy
    Submitted by: Alice Chen
    Submitted: Jan 21, 3:30 PM
    Status: Pending
    [View] [Approve] [Request Changes] [Reject]
```

Step 2: Review a Submission Click **"[View]"** to see the entry:

```
Database Indexing Strategy (SUBMISSION #45)


Submitted by: Carol Davis
Topic: organization/engineering/database/indexing
Submitted: Jan 22, 10:30 AM
References: 3 entries

## Overview

We're implementing a new indexing strategy to improve query performance...

[Full content displayed]

---

Reviewer Panel:


Status:  Pending Review

Your actions:
  [  Approve] [  Request Changes] [  Reject]
```

Step 3: Provide Feedback **Option A: Approve**

If the entry meets standards, click **"[Approve]"**:

```
Approve Submission


Comments (optional):
[Great job! Clear and well-referenced.]

[Approve]  [Cancel]
```

Option B: Request Changes

If you need revisions, click **"[Request Changes]"**:

```
Request Changes


Feedback (required):
```

```
[Please add a section on performance impact
and include benchmarks from testing.]

[Send Feedback] [Cancel]
```

The submitter gets notified and can revise.

Option C: Reject

If the entry doesn't fit, click **“[Reject]”**:

```
Reject Submission


Reason (required):
[Dropdown: Select reason...]
  • Out of scope for this notebook
  • Insufficient quality
  • Duplicates existing entry
  • Doesn't meet standards
  • Other

Comments:
[This topic is better suited for the Security notebook.
I'll forward them a reference.]

[Reject] [Cancel]
```

Step 4: Monitor Resubmissions After requesting changes, the queue updates:

```
Pending Submissions (2):


[ ] Caching Architecture (RESUBMISSION #2)
    Submitted by: Bob Johnson
    Originally submitted: Jan 22, 9:15 AM
    First feedback: "Needs more detail on consistency"
    Resubmitted: Jan 22, 2:00 PM
    Status: Awaiting review
    [View] [Approve] [Request Changes] [Reject]
```

View the updated submission, see what changed, and decide.

Review Criteria Examples

Example 1: Architecture Decision Record (ADR)

```
Good submission:
  - Clear problem statement
  - Decision and rationale
  - Consequences (positive and negative)
  - References related entries
  - Links to implementation

Poor submission:
  - Vague problem description
  - No rationale for why this decision
  - Doesn't address tradeoffs
  - No related references
```

Example 2: Technical Specification

```
Good submission:
  - Overview and motivation
  - Detailed specification with examples
  - Performance characteristics
  - Security considerations
  - API or configuration examples
  - Link to implementation

Poor submission:
  - "Here's our new API"
  - No examples
  - Doesn't explain why
  - Missing security analysis
```

Example 3: Incident Report

```
Good submission:
  - Timeline of events
  - Root cause analysis
  - Impact assessment
  - Mitigation steps taken
  - Preventive actions
  - Links to follow-up tasks

Poor submission:
  - "System went down"
  - No clear timeline
  - Blame focused vs. learning focused
  - No follow-up actions
```

Verification

Confirm review workflow is working:

- ☐ Submissions appear in review queue
- ☐ You can view submissions completely
- ☐ You can approve submissions (they're published)
- ☐ You can request changes (submitter is notified)
- ☐ You can reject submissions (recorded in audit log)
- ☐ Resubmissions after feedback are tracked

Tips & Tricks

Set Review Standards Document what you expect:

```
[Add to Notebook Description or FAQ]

Submission Standards:
  1. Clear, specific title
  2. Well-structured content (use headings)
  3. At least one reference to related entries
  4. Specific, not vague language
  5. Consider security implications
  6. No copyrighted content
```

Use Templates Provide templates for common entries:

```
[Create Template Entries]

Architecture Decision Record Template:
  - Problem Statement
  - Decision
  - Rationale
  - Consequences

Incident Report Template:
  - Timeline
  - Impact
  - Root Cause
  - Remediation
```

Fast-Track Approvals Don't require review for minor corrections:

```
Ingestion Gating


  Require review for new entries
  Require review for revisions to published entries
  Require review for minor fixes (typos, formatting)
```

Disable review for revisions to reduce bottlenecks.

Next Steps

After reviewing: - Provide constructive feedback - Publish approved entries - Help submitters improve rejected ones

---

Workflow 4: Monitoring Job Pipeline

Overview

Monitor background jobs (embeddings, claims analysis, comparisons) that process entries in your notebook. Jobs are created automatically; you just track them.

Use case: You want to see if background analysis is complete for entries your team just created.

Related workflows: - ThinkerAgent Configuration — Sets up agents that run these jobs

Prerequisites

- ☐ Notebook with entries
- ☐ At least "Read" access
- ☐ Understanding of job types (embeddings, claims, etc.)

Step-by-Step Instructions

Step 1: Access Job Statistics **Navigate to:** Notebooks → Select notebook → Statistics tab

```
Architectural Decisions


[Entry Feed] [Settings] [Statistics]

Job Queue Statistics:


Overall Status:  All caught up
Last updated: 5 minutes ago

Job Type           | Pending | In Progress | Completed | Failed

DISTILL_CLAIMS     | 0       | 0           | 1,247     | 0
COMPARE_CLAIMS     | 0       | 1           | 342       | 0
EMBED_ENTRIES      | 2       | 3           | 3,421     | 0
CLASSIFY_ENTRIES   | 0       | 0           | 4,892     | 0

Total entries processed: 9,902
Success rate: 99.97%

[Refresh Stats] [View Details] [Clear Failed]
```

Job Types:

Job	Purpose	Status
DISTILL_CLAIMS	Extract claims from entries	Should be completed
COMPARE_CLAIMS	Compare claims between entries	Should be completed
EMBED_ENTRIES	Create vector embeddings for search	Usually in progress
CLASSIFY_ENTRIES	Assign topics/categories	Background work

Step 2: View Detailed Job Status Click “[**View Details**]”:

```
Job Details - EMBED_ENTRIES


Pending (2):
  • entry_abc123 - "Database Indexing Strategy" (queued 5 min ago)
  • entry_def456 - "Caching Architecture" (queued 2 min ago)

In Progress (3):
  • entry_ghi789 - "API Versioning Policy" (processing 3 min)
  • entry_jkl012 - "Transaction Handling" (processing 1 min)
  • entry_mno345 - "Error Handling Standards" (processing < 1 min)

Completed (3,421):
  [Last 5 shown]
   entry_xyz999 - "Concurrency Model" (completed 2 min ago, 8s)
   entry_aaa111 - "Monitoring Architecture" (completed 5 min ago, 12s)
   entry_bbb222 - "Testing Strategy" (completed 8 min ago, 6s)

Failed (0):
  (none)
```

Step 3: Understand Job Timing Entries are processed in stages:

```
Entry Lifecycle:


1. Entry Created
   ↓ (immediately)
2. DISTILL_CLAIMS (extract claims)
   ↓ (1-2 minutes)
3. COMPARE_CLAIMS (compare to other entries)
   ↓ (1-2 minutes)
4. EMBED_ENTRIES (create vector embeddings)
   ↓ (1-2 minutes)
5. Ready for Search & Analysis

Typical total time: 5-10 minutes per entry
```

Step 4: Handle Failed Jobs If a job fails:

```
Failed (2):


  entry_xyz999 - "Concurrency Model"
   Job Type: EMBED_ENTRIES
   Failed: 10 minutes ago
   Error: "Timeout: embedding service unresponsive"
   [Retry] [View Error Log] [Dismiss]

  entry_aaa111 - "Monitoring Architecture"
   Job Type: COMPARE_CLAIMS
   Failed: 15 minutes ago
   Error: "Out of memory in comparison engine"
   [Retry] [View Error Log] [Dismiss]

[Retry All Failed] [Clear Failed] [Contact Support]
```

Click **"[Retry]"** to rerun the job:

```
Retrying: EMBED_ENTRIES for entry_xyz999

Status: Queued (will process in order)
[Cancel Retry]
```

The job will be re-queued and run again.

Step 5: Monitor Completion Jobs complete automatically. Monitor via the Statistics tab:

```
Checking every 5 minutes...

Jan 22, 3:00 PM: 5 pending → 2 pending (3 processed)
Jan 22, 3:05 PM: 2 pending → 0 pending (all complete!)

  All jobs complete for notebook!
```

Performance Insights

```
Job Performance Analysis


EMBED_ENTRIES Performance:
  Average time: 8.2 seconds per entry
  P99 time: 15 seconds
  Bottleneck: Vector database indexing

Success rate: 99.97% (1 failure in 3,421 jobs)
Most common error: Timeout (affects 0.03%)

Recommendation:
  Bottleneck is in vector DB. Consider:
    • Increasing DB connection pool
    • Scaling embedding service
```

Verification

Confirm job monitoring is working:

☐ You can see job queue statistics
☐ Job counts add up (pending + in progress + completed)
☐ You can view detailed job information
☐ Failed jobs can be retried
☐ Completion rate is tracked
☐ Performance metrics are available

Tips & Tricks

Auto-Refresh Dashboard Set up auto-refresh while monitoring:

```
[Auto-Refresh] [Every 5 minutes]

Or set polling interval:
    Never
    Every minute
    Every 5 minutes
    Every 10 minutes
```

Understand Stalls If jobs aren't progressing:

```
  Why are jobs stuck in "In Progress"?

Check:
  1. Agent status - Is the processing agent active?
  2. Agent logs - Are there errors?
  3. System health - CPU/memory/disk OK?
  4. Network - Can agent reach Cyber?
  5. Job logs - Specific error message?
```

Scale Based on Load Monitor job backlog:

```
High backlog (100+ pending)?
  → You may need more agents
  → Consider parallel processing
  → Talk to System Admin about scaling

No backlog (< 5 pending)?
  → Current capacity is sufficient
  → Don't add more agents unnecessarily
```

Next Steps

After monitoring: - Investigate failed jobs - Understand performance bottlenecks - Request agent scaling if needed

---

Workflow 5: Managing Subscriptions

Overview

Subscribe your notebook to other notebooks to mirror entries and keep knowledge synchronized across your organization or even across organizations.

Use case: Your team uses insights from another team's research. You subscribe to their notebook to automatically mirror new entries.

Related workflows: - Cross-Organization Coordinator — Managing subscriptions at org level

Prerequisites

☐ Source notebook you want to subscribe to (you have Read access)
☐ Admin access to your notebook
☐ Understanding of subscription scope and filtering

Step-by-Step Instructions

Step 1: Go to Subscriptions **Navigate to:** Notebooks → Select notebook → Subscriptions tab

```
Architectural Decisions


[Entry Feed] [Settings] [Subscriptions]

Active Subscriptions (1):



[Source] Infrastructure / Database Design
  Scope: Entries (catalog + claims + entries)
  Synced: 45 entries (last 2 hours)
  Status:  Healthy
  Watermark: Position 1,247
  [View] [Sync Now] [Pause] [Edit] [Unsubscribe]

[+ Subscribe to Notebook]
```

Click “[+ **Subscribe to Notebook**]”.

Step 2: Select Source Notebook

```
Subscribe to Notebook


Find source notebook:
[Search or select...] [Browse organizations]

Recent notebooks:
 Infrastructure / Database Design
 Security / Incident Response
 Operations / Runbooks

Organizations:
  MyCompany
     Engineering / Architecture
     Engineering / Backend
     Operations / Runbooks
  OtherCompany (partner org)
     Public / Documentation
     Public / Standards
```

Search or browse to find the notebook you want to subscribe to.

Step 3: Configure Subscription Scope

```
Subscription Settings


Source Notebook: Infrastructure / Database Design

Scope *
[Dropdown: What to mirror...]

 Catalog only     (titles and metadata)
 Catalog + Claims (titles + extracted claims)
 Entries          (full entries, catalog, claims)

Discount Factor
[Slider: 100%] ← How much to weight new entries

  100% = Full relevance
  50%  = Half weight in coherence calculations
  10%  = Low relevance (reference only)

Polling Interval
[Dropdown: How often to check...]

   Every hour
   Every 4 hours
   Every day
   Manual only

[Subscribe] [Preview] [Cancel]
```

Scope Options:

Scope	What You Get	Use Case
Catalog	Entry titles, metadata, topics	Quick reference
Catalog + Claims	Above + extracted claims	Analysis, comparison

Scope	What You Get	Use Case
Entries	Full content + claims + metadata	Deep integration, learning

Discount Factor: - 100% = These entries are just as relevant as local ones - 50% = These entries are somewhat relevant (external perspective) - 10% = These entries are reference-only (not central to us)

The discount affects integration cost calculation—external entries don't override local consensus.

Step 4: Subscribe Click "[**Subscribe**]":

```
  Subscription created!

Source: Infrastructure / Database Design
Entries mirrored: 0 (first sync in progress...)
Status: Syncing...

Next sync: In 4 hours (or on schedule)

You can:
  [View Mirrored Entries] [Sync Now] [Manage Subscription]
```

Step 5: Monitor Sync Status Your subscription dashboard shows sync progress:

```
Subscriptions Dashboard


Infrastructure / Database Design


Sync Status:  Healthy (last sync: 2 hours ago)
Mirrored: 45 entries
Watermark: Position 1,247 (45/45 synced)
Next sync: In 2 hours

Errors (last 7 days): 0
Skipped entries: 0 (all entries accessible)

[View Mirrored Entries] [Sync Now] [Edit Settings] [Unsubscribe]
```

Step 6: View Mirrored Entries Mirrored entries appear in your notebook marked as external:

```
Entry Feed


[Entry] Query Optimization Patterns
  Author: Alice Chen (Infrastructure Team) → External
  Source: Infrastructure / Database Design
  Position: [external-sync-1247]
  Integration:  Probation (mirrored, discount 50%)
  [Read] [View Source] [Remove from Local Copy]
```

Click "[**View Source**]" to go to the original entry.

Subscription Scenarios

Scenario 1: Internal Cross-Team Subscription

```
Backend team subscribes to Infrastructure team's database decisions

Scope: Entries (full details)
Discount: 100% (internal, equally relevant)
Polling: Every 4 hours
```

```
Backend engineers can:
   Learn from infrastructure decisions
   Reference infrastructure entries
   Understand database patterns
```

Scenario 2: Cross-Organization Research

```
Your research team subscribes to partner org's public research

Scope: Catalog + Claims (detailed)
Discount: 50% (external, useful reference)
Polling: Every day

Your team can:
   Know what partners are researching
   Avoid duplicate work
   Build on partner's findings
   Risk: Some entries may not apply to your context
```

Scenario 3: Regulatory Standard Reference

```
Your compliance team subscribes to standards organization's guidelines

Scope: Catalog only (just reference)
Discount: 10% (external standard, low discount)
Polling: Manual only (standards rarely change)

Compliance can:
   Reference official standards
   Link entries to compliance requirements
   Entries don't influence local coherence
```

Verification

Confirm subscription is working:

- □ Subscription appears in your subscriptions list
- □ Mirrored entries appear in entry feed
- □ Entries marked as external/mirrored
- □ Sync status shows healthy
- □ Watermark is advancing (being synced)
- □ Can view original entry via link

Tips & Tricks

Manual Sync When Urgent Force a sync without waiting for schedule:

```
[Sync Now]

Status:  Syncing...
New entries since last sync: 3
Syncing: [     ] 80% complete
```

Control Over Local Copies After mirroring, you can edit the local copy:

```
Mirrored Entry: Query Optimization Patterns

[ Create Local Copy]

This creates an editable version in your notebook that:
  • Can be revised independently
  • Still links to the original
  • Appears in your search results
```

Selective Subscription Subscribe to specific topics only:

```
[Advanced Settings]

Topic Filter:
[Include topics matching...]
   infrastructure/database
   infrastructure/performance
   infrastructure/security

Only mirror entries matching these topics.
```

Conflict Resolution If you edit a mirrored entry and it changes in the source:

```
  Update available for "Query Optimization Patterns"

Local version: Position [local-1234]
Source version: Position [external-1247] (newer)

Differences:
  - Source added: "Include compound indexes"
  - Source changed: "Performance impact: +15%"

Options:
  [Keep Local Version] [Merge Changes] [Use Source Version]
```

Next Steps

After subscribing: - Review mirrored entries for relevance - Reference them in your entries - Periodically review subscription health

---

Summary: Quick Reference

The 5 Workflows at a Glance

Workflow	Purpose	Time	Frequency
1. Create Notebooks	Set up knowledge space	15-30 min	Quarterly
2. Manage Access	Grant/revoke permissions	5-15 min	As needed
3. Review Submissions	Approve/reject entries	10-30 min	Continuous
4. Monitor Jobs	Track background processing	5-10 min	Daily
5. Manage Subscriptions	Mirror external notebooks	10-20 min	As needed

Your Workflow Loop

```
1. Create Notebook (once)
   ↓
2. Manage Access (ongoing)
   ↓
3. Enable Review Gates (optional)
   ↓
4. Monitor Jobs (daily)
   ↓
5. Manage Subscriptions (quarterly)
   ↓
6. Back to Step 2 (continuous)
```

Key Responsibilities

- **Security:** Classify appropriately, manage access, audit changes
- **Quality:** Review submissions, maintain standards, organize knowledge
- **Operations:** Monitor job health, fix failures, manage subscriptions
- **Collaboration:** Grant access to stakeholders, enable cross-team learning

---

Related Personas

Your workflows overlap with:

- **Knowledge Contributor** — Who create and submit entries
- **Organization Administrator** — Who set up organizational structure
- **Auditor/Compliance Officer** — Who review your notebook for compliance
- **System Administrator** — Who manage platform-wide agents and monitoring

---

Troubleshooting

Can't Create Notebook

Cause: You don't have "Admin" access to an owner group.

Solution: 1. Check your group memberships (Settings → Profile → Groups) 2. Request admin role in a group from your organization admin 3. Or create a new group with your organization admin's help

Access Control Not Taking Effect

Cause: Clearance cache or permission propagation delay.

Solution: 1. Wait 5 minutes for changes to propagate 2. Admin → Organizations → Flush Clearance Cache 3. User logs out and back in

Job Stuck in "In Progress"

Cause: Processing agent is down or overloaded.

Solution: 1. Check agent status in Admin → Agents 2. Check agent logs for errors 3. If agent is down, restart it 4. Retry job from the Job Details view

Subscription Sync Failing

Cause: Source notebook permissions changed or source notebook deleted.

Solution: 1. Check you still have Read access to source notebook 2. Verify source notebook still exists 3. Check network connectivity to source org 4. Edit subscription and re-test connection

Can't Edit Mirrored Entry

Cause: It's a read-only external reference.

Solution: 1. Click "[Create Local Copy]" to make an editable version 2. Edit the local copy (still links to original) 3. Keep both versions in sync manually or via subscription

---

Last updated: February 21, 2026 **Chapter version:** 1.0.0 (Beta) **Platform version:** 2.1.0

- Query and analyze audit logs
- Investigate security events and access denials
- Monitor compliance with security policies
- Generate audit reports for regulators
- Track data retention and classification compliance
- Review cross-organization information flows

Required Permissions: - “Admin” access to audit logs - Your organization’s clearance (at least SECRET recommended) - Understanding of security model and compliance requirements

Typical Workflows: 3 core workflows in this chapter

Workflow 1: Querying Global Audit Logs

Overview

Access and filter the organization-wide audit log to see who did what, when, and to which resources.

Use case: Your compliance team needs to generate a quarterly audit report showing all access to sensitive data.

Related workflows: - Investigating Security Events — Deep dive into specific incidents - Notebook-Scoped Auditing — Focused audits on specific notebooks

Prerequisites

- ☐ Audit admin role in your organization
- ☐ Clear understanding of what you’re looking for
- ☐ Time range for audit query
- ☐ Optional: Specific users/resources to filter

Step-by-Step Instructions

Step 1: Access Audit Log **Navigate to:** Admin Panel → Audit Log (or Admin → Organizations → [Your Org] → Audit Log)

```
Global Audit Log


[Filters] [Search] [Export to CSV] [Generate Report]

Filters:
  Actor:        [All users  ]
  Action:       [All actions  ]
  Resource:     [All resources  ]
  Date Range:   [Jan 1 - Jan 31, 2026]
  Status:        Success    Failure    Denied

Results: 1,247 events

Entry Feed (sorted by newest first):


Timestamp      | Actor    | Action    | Resource         | Status

Jan 31, 2:30 PM | Jane S.  | WRITE     | nb_xyz/entry_123 |   OK
Jan 31, 2:15 PM | Bob J.   | READ      | nb_abc/entry_456 |   OK
Jan 31, 1:45 PM | Alice C. | REVISE    | nb_xyz/entry_789 |   OK
Jan 31, 1:30 PM | Carol D. | READ      | nb_secret/...    |   DENIED
Jan 31, 1:00 PM | Eve W.   | SHARE     | nb_xyz           |   OK
```

Step 2: Apply Filters **Actor Filter:** Search for specific users

```
Actor Filter:
[Type name or select...]

Results:
   Alice Chen (alice@company.com)
   Bob Johnson (bob@company.com)
   Carol Davis (carol@company.com)
   David Smith (david@company.com)
```

```
    All ThinkerAgents
    System (internal actions)
```

Check one or more users to filter logs.

Action Filter: Filter by operation type

```
Action Type:
    WRITE (create entries)
    REVISE (update entries)
    READ (view entries)
    SHARE (grant access)
    DELETE (remove entries)
    ADMIN (manage notebook)
```

Resource Filter: Filter by notebook/entry

```
Resource:
[Type notebook name...]

Results:
    Engineering / Architecture (nb_eng_arch)
    Operations / Runbooks (nb_ops_runbooks)
    Security / Incidents (nb_sec_incidents)
```

Date Range: Set audit period

```
From: [Jan 1, 2026] To: [Jan 31, 2026]

Presets:
    Last 7 days
    Last 30 days (default)
    Custom range
```

Status Filter: Include/exclude results

```
  Success (operations that succeeded)
  Failure (operations that failed for technical reasons)
  Denied (access control denials)

Example: Uncheck "Denied" to see only successful operations
```

Step 3: Examine Audit Events Each audit event shows:

```
Jan 31, 2:30 PM - Jane Smith accessed Engineering/Architecture

Action:     WRITE
Status:       Success
Resource:   Notebook: nb_eng_arch, Entry: entry_abc123
Actor:      Jane Smith (auth_hash_xyz)
Timestamp:  Jan 31, 2026, 2:30 PM UTC
IP Address: 192.168.1.50
User Agent: Chrome 120.0 / macOS
Location:   San Francisco, US (GeoIP)

Details:
  Entry Title: "Microservices Architecture Decision"
  Entry Topic: organization/engineering/architecture
  Signature:   Valid (Ed25519 signature verified)
  Clearance Used: SECRET / {Operations}

[View Entry] [Related Events for Jane Smith] [View Similar Actions]
```

Step 4: Investigate Anomalies Look for suspicious patterns:

```
Anomaly Indicators:
   Carol Davis accessed 47 entries in 5 minutes
    (normal: 2-3 per hour)
    Action: [Investigate] [Allowlist Pattern]

   System (internal) failed to embed 12 entries
    (retry pattern detected)
    Action: [View Error Details] [Notify Admin]

  Alice Chen denied access to TOP_SECRET notebook 3 times
    (clearance mismatch)
    Action: [Review Clearance] [Contact Alice]
```

Verification

Confirm your audit query is complete:

- ☐ Correct date range selected
- ☐ Filters applied appropriately
- ☐ All relevant events retrieved
- ☐ No suspicious patterns missed
- ☐ Audit trail is uninterrupted (no gaps)

Tips & Tricks

Export for Reporting Click "[**Export to CSV**]" to download results:

```
audit_log_2026-01_31.csv

timestamp,actor,action,resource,status,ip_address,location
2026-01-31T14:30:00Z,Jane Smith,WRITE,nb_eng_arch/entry_abc123,success,192.168.1.50,San Francisco
2026-01-31T14:15:00Z,Bob Johnson,READ,nb_abc/entry_456,success,10.0.0.5,New York
...
```

Use in Excel/Google Sheets for further analysis.

Generate Compliance Report Click "[**Generate Report**]" for automated output:

```
Compliance Audit Report - January 2026


Executive Summary:
  Total Events: 47,329
  Success Rate: 99.2%
  Denied Accesses: 384 (0.8%)
  Critical Incidents: 0

Access Denial Analysis:
  Reason: Clearance Insufficient - 287
  Reason: Entry Not Found - 64
  Reason: Notebook Access Denied - 33

Top Accessed Resources:
  1. Engineering/Architecture: 12,483 accesses
  2. Operations/Runbooks: 8,923 accesses
  3. Security/Incidents: 4,521 accesses

Recommendations:
  • Review Carol Davis's clearance (accessing 15% of all entries)
  • Investigate 12 failed embedding jobs in EMBED_ENTRIES
  • Verify access to TOP_SECRET entries (47 accesses, 3 denials)
```

Real-Time Monitoring Set up continuous monitoring for specific patterns:

```
[Create Alert]

Alert Name: Unusual Access Pattern

Trigger Condition:
  Same user accesses > 50 entries in < 1 hour
  AND entries are in different topics
  AND user's normal pattern is 5-10 per hour

Actions:
   Send notification
   Log to compliance queue
   Automatically disable account (don't recommend)

[Save Alert]
```

Workflow 2: Investigating Security Events

Overview

Deep-dive investigation when you detect access denials, unusual patterns, or suspected policy violations.

Use case: Multiple failed access attempts to a TOP_SECRET notebook. You investigate to determine if it's a misconfiguration or a security incident.

Related workflows: - Querying Global Audit Logs — Find the events - Notebook-Scoped Auditing — Focused audit

Prerequisites

- ☐ Audit admin role
- ☐ Specific event or pattern to investigate
- ☐ Access to security logs and incident reporting system

Step-by-Step Instructions

Step 1: Identify Suspicious Events From the audit log, find events matching one of these patterns:

```
 Red Flags:
 • Multiple DENIED events from one user (attempted breach?)
 • Unusual volume (Carol accessed 100 entries in 30 min)
 • Off-hours access (access at 3 AM on Sunday)
 • Access to mismatched topics (why is developer accessing HR files?)
 • Privilege escalation (user suddenly accessing TOP_SECRET)
 • Failed operations (500+ failed embeds in one hour)
```

Step 2: View Detailed Event Click on a suspicious event for full details:

```
Investigation: Access Denial - TOP_SECRET Data

Primary Event:



Timestamp: Jan 31, 2:15 PM
Actor: Alice Chen (alice@company.com)
Action: READ
Resource: Notebook "Security / TOP_SECRET Planning"
Status:  DENIED

Denial Reason:
  Clearance Insufficient
  Required: TOP_SECRET / {Operations, Strategic}
  User has: SECRET / {Operations}
  Gap: Missing TOP_SECRET level + Strategic compartment

User Context:


Groups: Engineering, Backend Team, Project Alpha
Clearance: SECRET / {Operations}
Last clearance change: 3 months ago
Previous denied accesses: 0 (first time)

IP/Session Context:


IP Address: 192.168.1.200
Location: San Francisco, US (matches normal location)
Device: Chrome 120 / macOS (matches normal device)
Session: New session (5 minutes old)
VPN: Not detected

Related Events:


[View all events for Alice Chen in last 7 days]
[View all accesses to this notebook in last 7 days]
[View all failed accesses to TOP_SECRET resources]
```

Step 3: Make Determination Based on investigation, determine incident classification:

```
Incident Classification


Incident Type:
   False Positive (legitimate, permission issue)
   Policy Violation (user bypassed or exceeded permissions)
   Misconfiguration (system assigned wrong clearance)
   Security Incident (unauthorized access attempt)
   Suspicious Activity (needs investigation)

Severity (if applicable):
   Low (informational)
   Medium (policy question)
   High (confirmed violation)
   Critical (security breach)

Root Cause Analysis:
[Alice was recently promoted but clearance wasn't updated.
 She tried to access materials for her new role.]

Recommendation:
[Promote Alice to TOP_SECRET / {Operations, Strategic}
 clearance and notify her of successful access.]

[Log Finding] [Close Incident] [Escalate to Security]
```

Step 4: Take Action Based on determination, take appropriate action:

If False Positive:

```
[ Resolve Incident - Permission Issue]
```

```
Actions taken:
  • Grant Alice TOP_SECRET clearance
  • Flush clearance cache
  • Verify access now works
  • Log resolution for compliance

Next: Verify access works, close incident.
```

If Policy Violation:

```
[ Resolve Incident - Policy Violation]

Actions taken:
  • Document violation in policy log
  • Notify user's manager
  • Review similar events
  • Update access controls if needed

Next: Follow up with manager.
```

If Security Incident:

```
[ ESCALATE - Security Incident]

Actions:
  • Lock user account (require immediate review)
  • Notify Security Team immediately
  • Preserve all related logs
  • Initiate incident response

Next: Contact Security Operations Center.
```

Verification

Confirm investigation is thorough:

☐ Identified root cause
☐ Checked for related events
☐ Verified user context (location, device, patterns)
☐ Determined if isolated or pattern
☐ Documented findings
☐ Took appropriate action

Workflow 3: Notebook-Scoped Auditing

Overview

Audit a specific notebook to verify compliance with its policies, review access patterns, and track data handling.

Use case: Quarterly compliance review of the "TOP_SECRET Strategic Planning" notebook. You need to verify who accessed it, what they did, and if any policy violations occurred.

Related workflows: - Querying Global Audit Logs — Org-wide audits - Investigating Security Events — Deep investigation

Prerequisites

☐ At least "Read" access to the notebook
☐ Audit or admin role
☐ Clear compliance requirements

Step-by-Step Instructions

Step 1: Access Notebook Audit Trail **Navigate to:** Notebooks → Select notebook → Audit tab

```
Engineering / Architecture Notebook


[Entry Feed] [Settings] [Audit] [Statistics]

Notebook Audit Trail


Classification: SECRET / {Operations}
Owner: Engineering / Backend Team
Created: Jan 1, 2026, 9:00 AM (by Alice Chen)
Last Modified: Jan 31, 2026, 2:30 PM

Access Summary (Last 30 days):
  Total reads: 1,247
  Total writes: 89
  Total revisions: 23
  Total admin actions: 12
  Access denials: 0

Detailed Audit Log:


[Filters] [Export] [Generate Report]

Timestamp      | Actor      | Action     | Details           | Status

Jan 31, 2:30 PM | Jane S.    | WRITE      | New entry created |
Jan 31, 2:15 PM | Bob J.     | READ       | 15 entries read   |
Jan 31, 1:45 PM | Alice C.   | REVISE     | Entry updated     |
Jan 31, 1:30 PM | Carol D.   | ADMIN      | Access granted    |
Jan 31, 1:00 PM | Eve W.     | SHARE      | Group added       |
```

Step 2: Review Access Control Changes Track who has access and when it changed:

```
Access Control Changes (Last 30 days):


Jan 31, 1:30 PM - Carol Davis granted "Operations Team" Read+Write access
  Granted by: Alice Chen
  Reason: Team needs visibility for incident response

Jan 28, 10:00 AM - Contractor "David Smith" removed from Read+Write
  Removed by: Alice Chen
  Reason: Contract ended

Jan 15, 2:00 PM - "Executive Council" granted Read access
  Granted by: Alice Chen
  Reason: Quarterly review attendance

Changes Summary:
   All changes documented with reasons
   All grantors are notebook admins
   No orphaned access (all removals justified)
   Access levels appropriate for roles
```

Step 3: Review Data Lifecycle Track entries created, modified, and retained:

```
Entry Lifecycle Audit


Entries Created: 89 (month-to-date)
```

```
  Average per day: 2.9
  Range: 1-7 entries per day
  Busiest day: Jan 21 (7 entries)

Entries Revised: 23
  Revision rate: 25.8% of entries (1 in 4 has revision)
  Average revisions per entry: 1.3
  Longest history: 4 revisions

Entries Deleted: 0
  Retention policy: 7 years
  Next purge eligible: None

Data Classification Compliance:
   100% of entries labeled SECRET / {Operations}
   0 entries with inconsistent classification
   0 entries with higher classification (not breached)
   All entries have external references checked
```

Step 4: Generate Compliance Report Click “[**Generate Report**]”:

```
Notebook Compliance Report - January 2026


Notebook: Engineering / Architecture
Classification: SECRET / {Operations}
Report Period: January 1-31, 2026
Generated: Jan 31, 2026, 3:00 PM

EXECUTIVE SUMMARY


Compliance Status:  COMPLIANT
  • All entries properly classified
  • Access control is appropriate
  • No policy violations detected
  • All changes documented

DETAILED FINDINGS


Access Control:
  Approved Users: 12
  Approved Groups: 3
  Denied Accesses: 0
  Clearance Mismatches: 0
   PASSED

Data Classification:
  Total Entries: 89
  Correct Classification: 89/89 (100%)
  Misclassified: 0
   PASSED

Entry Lifecycle:
  Retention Policy: 7 years
  Eligible for Purge: 0 entries
  Average Version Count: 1.3
   PASSED

RECOMMENDATIONS
```

```
1. Continue current access practices (working well)
2. Monitor revision patterns (stable at 25.8%)
3. Review contractor removals monthly (currently quarterly)

SIGN-OFF


Auditor: Jane Smith (Compliance Officer)
Date: January 31, 2026
Signature: [Digital signature verified]

[Download PDF] [Email Report] [Acknowledge Audit]
```

Verification

Confirm notebook audit is complete:

- ☐ Reviewed all access control changes
- ☐ Verified data classification
- ☐ Checked entry lifecycle
- ☐ Examined revision patterns
- ☐ Generated compliance report
- ☐ Documented findings

Tips & Tricks

Automate Compliance Reviews Set up recurring audits:

```
[Schedule Recurring Audit]

Notebook: Engineering / Architecture
Frequency: Monthly (last day of month)
Recipients: compliance@company.com
Report Type: Abbreviated (key metrics only)

[Save Schedule]
```

Compare Year-Over-Year Track trends:

```
Access Pattern Trends


Total Reads per Month:
  Jan 2025: 847   Jan 2026: 1,247 (+47%)
  Feb 2025: 921   Feb 2026: (projected 1,300+)

Entry Creation Rate:
  Jan 2025: 42 entries   Jan 2026: 89 entries (+112%)

Revision Rate:
  Jan 2025: 18% of entries
  Jan 2026: 26% of entries (more collaborative)

Interpretation: Notebook growing in usage and collaboration.
Recommendation: Consider archiving to separate "historical" notebook.
```

Bulk Export for Compliance Export all audit logs for external auditors:

```
[Bulk Export - Last 12 Months]

Format: CSV
```

```
Period: Jan 1 - Dec 31, 2025
File: notebook_audit_2025.csv (2.3 MB)

Columns included:
  - timestamp, actor, action, resource, status
  - ip_address, location, clearance_used
  - entry_classification, entry_topic
  - signature_valid, details

[Download] [Email to Auditor] [Encrypt & Send]
```

Summary: Quick Reference

The 3 Workflows at a Glance

Workflow	Purpose	Time	Frequency
1. Query Logs	Find audit events	15-30 min	Quarterly
2. Investigate	Deep dive on incidents	30-60 min	As needed
3. Notebook Audit	Compliance review	20-40 min	Monthly

Your Audit Loop

```
1. Query Global Logs (baseline)
   ↓
2. Find Anomalies
   ↓
3. Investigate if needed
   ↓
4. Generate Reports
   ↓
5. Follow up on findings
```

Audit Focus Areas

- **Access Control:** Who has access? Is it appropriate?
- **Classification:** Are entries labeled correctly?
- **Lifecycle:** Are entries retained/purged per policy?
- **Changes:** Are all modifications authorized and logged?
- **Incidents:** Are security incidents handled properly?

Related Personas

Your workflows overlap with:

- **System Administrator** — Who manage platform-wide security
- **Knowledge Contributor** — Whose access you audit
- **Notebook Owner** — Who manage notebooks you audit

Troubleshooting

Can't Access Audit Logs

Cause: Don't have audit admin role

Solution: 1. Request audit admin role from your organization admin 2. Verify you're in the right organization 3. Check if role is limited to specific notebooks

Audit Logs Show Gaps

Cause: Log rotation or system maintenance

Solution: 1. Check system status page for known outages 2. Verify your date range is correct 3. Contact admin if gaps are suspicious

Export File Too Large

Cause: Exporting too much data at once

Solution: 1. Narrow date range 2. Filter by specific actor/resource 3. Use CSV format (smaller than JSON) 4. Export in batches by date

---

Last updated: February 21, 2026 **Chapter version:** 1.0.0 (Beta) **Platform version:** 2.1.0

- Manage user accounts globally (create, lock, unlock, delete)
- Set and enforce usage quotas
- Monitor system health and performance
- Manage ThinkerAgent deployment and configuration
- Review platform-wide security settings
- Handle user support and account issues

Required Permissions: - “Admin” role (platform-level, not organization-level) - ROOT or SUPERUSER clearance (highest available) - Understanding of system architecture and operations

Typical Workflows: 4 core workflows in this chapter

Workflow 1: User Management

Overview

Create user accounts, manage permissions, lock/unlock accounts, and handle user lifecycle.

Use case: New employee joins; you create their account and set initial permissions. Later, they leave and you deactivate their account.

Related workflows: - Quota Management — Set usage limits - Agent Management — Grant agent access

Prerequisites

- ☐ System admin access (ROOT/SUPERUSER clearance)
- ☐ User information (email, organization, initial clearance)
- ☐ Clear policy for account creation

Step-by-Step Instructions

Step 1: Access User Management **Navigate to:** Admin Panel → Users (or Settings → System → Users)

```
User Management


[+ Create User] [Import Users] [Export Users]

Active Users: 147

Search/Filter:
  [Search by email...] [Organization ] [Status ]

User List:


Email                       | Org       | Clearance       | Status   | Actions
```

```
alice@company.com        | MyCompany | SECRET/{Ops}   | Active  | [Edit]
bob@company.com          | MyCompany | CONF/{Ops}     | Active  | [Edit]
carol@partner.org        | Partner   | CONF/{}        | Active  | [Edit]
david@company.com        | MyCompany | SECRET/{Ops}   | Locked  | [Unlock]
```

Click “[+ **Create User**]”.

Step 2: Create User Account

```
Create User Account


Basic Information:
  Email *: [new.user@company.com]
  Full Name: [Jane Smith]
  Organization *: [Dropdown: Select...]
      MyCompany
      Partner Org
      Contractor Org

Initial Clearance:
  Level *: [CONFIDENTIAL  ]
  Compartments: [Select compartments...]
      Operations
      Strategic Planning
      Database Access

User Type:
    Human User (standard)
    Service Account (for automation)
    Bot/Agent (see Agent Management)

Sending Options:
    Send activation email
    Send temporary password (if password auth)

[Create Account] [Cancel]
```

Step 3: Manage Active User To view or edit an existing user, click “**[Edit]**”:

```
Edit User: Jane Smith


Profile:
  Email: jane@company.com
  Full Name: Jane Smith
  Organization: MyCompany
  Created: Jan 1, 2026
  Last Login: Jan 31, 2026, 2:30 PM

Clearance:
  Current: SECRET / {Operations, Database}
  Update: [Change Clearance  ]

Account Status:
    Active (can log in)
    Locked (cannot log in, but account exists)
    Disabled (account deleted, data archived)

Quota Usage:
  Notebooks Created: 3/5 (60%)
  Entries Written: 245/1000 (24.5%)
  Storage Used: 12.5 MB / 1 GB (1.25%)
  [View Detail] [Reset Quotas]

Actions:
  [Lock Account] [Unlock Account] [Reset Password]
  [View Audit Trail] [Delete Account]

[Save Changes] [Cancel]
```

Step 4: Lock/Unlock Account If a user forgets their password or has security issues:

```
[Lock Account]

Reason for locking:
[Dropdown: Select...]
  • User forgot password
  • Security incident investigation
  • Account compromise suspicion
  • User on leave
  • Other (specify)

Notify user?
    Send notification that account was locked
    Silent lock (for security incidents)

[Confirm Lock]
```

User can't log in but account/data remain intact.

To reactivate:

```
[Unlock Account]

Reason for unlocking:
[Dropdown: Select...]
  • Password reset complete
  • Investigation cleared
  • User returned from leave
  • Other

Send temporary password?
    Send new temporary password
    User will use existing password

[Confirm Unlock]
```

Step 5: Bulk User Management Import multiple users at once:

```
[Import Users]

File Format: CSV
[Upload file: users_batch_jan2026.csv]

Preview:
  email,organization,clearance_level,compartments,user_type
  alice@company.com,MyCompany,SECRET,Operations;Database,human
  bob@company.com,MyCompany,CONFIDENTIAL,Operations,human
  carol@company.com,MyCompany,CONFIDENTIAL,,human

Validation:
    3 rows ready to import
    All emails valid
  All organizations exist
  All clearances valid

[Preview Changes] [Import] [Cancel]
```

Verification

Confirm user management is working:

- ☐ New users can log in
- ☐ Clearances are correct
- ☐ Quotas are initialized
- ☐ Locked accounts can't access
- ☐ Unlock restores access
- ☐ Audit trail records all changes

Workflow 2: Quota Management

Overview

Set and monitor per-user quotas for notebooks, entries, and storage to prevent resource exhaustion.

Use case: A heavy user is approaching their storage quota. You increase it to prevent disruption.

Related workflows: - User Management — Create users with quotas - System Monitoring — Monitor quota usage

Prerequisites

- ☐ System admin access
- ☐ User to update quotas for
- ☐ Clear policy for quota limits

Step-by-Step Instructions

Step 1: Access Quota Management **Navigate to:** Admin → Quotas (or Users → [User] → Quotas)

```
Quota Management


Default Organization Quotas:
  Notebooks per user: 10
  Entries per notebook: 10,000
  Storage per user: 1 GB
  API calls per day: 10,000

[+ Create Custom Quota] [Reset to Defaults]

User Custom Quotas:


Search user: [alice@company.com]

[Edit Quotas]
```

Step 2: Set User Quotas

```
Quotas for: Alice Chen


Default (for all other users):
  Notebooks: 10
  Entries per notebook: 10,000
  Storage: 1 GB

Alice's Custom Quotas:
  Notebooks:
    Default: 10    Adjusted: [25] (she manages multiple teams)

  Entries per notebook:
    Default: 10,000   Adjusted: [50,000] (large-scale project)

  Storage:
    Default: 1 GB     Adjusted: [5 GB] (research data)

  API calls per day:
    Default: 10,000   Adjusted: [50,000] (integrations)

Effective Quotas (after changes):
   Notebooks: 25
   Entries: 50,000 per notebook
   Storage: 5 GB total
   API: 50,000 calls/day

Justification:
[Alice manages 8 cross-functional projects requiring
 heavy data management and automated integrations.]

[Save Quotas] [Cancel] [Reset to Defaults]
```

Step 3: Monitor Quota Usage View current usage for a user:

```
Quota Usage: Alice Chen


Notebooks:
  Used: 18/25 (72%)       [Close to limit]
  Recent: Created "Marketing Q2" on Jan 31
  Action: [Warn User] [Increase Quota]

Entries per Notebook:
  Max Used (across notebooks): 8,247/50,000 (16%)
  Notebook: "Q1 Planning" has 8,247 entries
  Action: [No action needed]

Storage:
  Used: 4.2 GB / 5 GB (84%)       [Close to limit]
  Recent uploads: 450 MB in last 7 days
  Projected: Will exceed limit in ~3 days
  Action: [Increase Quota] [Warn User] [Archive Entries]

API Calls:
  Used (today): 32,456 / 50,000 (65%)
  Average daily: 28,000
  Peak: 47,000 (Jan 29)
  Action: [No action needed]

[Adjust Quotas] [Notify User] [Archive Entries]
```

Step 4: Enforce Quotas When quotas are exceeded:

```
Quota Exceeded: Alice Chen

Storage limit reached (5 GB / 5 GB)
```

```
Actions available:
  [Increase Quota] (recommended for active users)
  [Archive Old Entries] (compress and move to archive)
  [Delete Entries] (permanent, unreversible)
  [Lock Account] (last resort, prevent more writes)

Current enforcement: Warning (writes still allowed)
Options:
    Warning only (user can still write)
    Block new writes (force action)
    Lock account (emergency)

[Apply] [Notify User] [Cancel]
```

Verification

Confirm quota management is working:

- ☐ Default quotas set appropriately
- ☐ Custom quotas applied to heavy users
- ☐ Usage monitored and alerted
- ☐ Exceeded quotas enforced
- ☐ Users notified of limits

Workflow 3: System Monitoring

Overview

Monitor platform health, performance, and usage. Review dashboards and metrics to ensure system stability.

Use case: You notice slow performance; you check the dashboard and find a ThinkerAgent is down, causing job backlog.

Related workflows: - Agent Management — Restart agents - Quota Management — Monitor resource usage

Prerequisites

- ☐ System admin access
- ☐ Understanding of system metrics
- ☐ Access to alerting system

Step-by-Step Instructions

Step 1: Access Dashboard **Navigate to:** Admin → Dashboard (or Home → System Status)

```
System Dashboard


System Health:  All Systems Operational
Last Updated: 31 Jan 2026, 3:30 PM

Overall Metrics:
  Uptime: 99.97% (last 30 days)
  Response Time: 145ms average
  Error Rate: 0.003% (< 1 per 100,000 operations)

Active Users: 147 (69 in last 24 hours)
Notebooks: 389 (12 created this week)
Entries: 18,392 (145 added today)

Server Status:
```

```
API Server:       Healthy (98% CPU, 72% Memory)
Database:         Healthy (replication lag: 50ms)
Cache Layer:      Healthy (hit rate: 94%)
Job Queue:         SLOW (backlog: 234 jobs, 12 failed)
Search Index:     Healthy (updated 5 minutes ago)
```

Step 2: Investigate Issues Click on the symbol for more details:

```
Job Queue - Performance Issue


Status:   DEGRADED (high backlog)

Job Statistics:
  Pending: 234 jobs (normal: 5-10)
  In Progress: 0 jobs (agents not processing!)
  Completed: 12,847
  Failed: 12 (since midnight)

Failed Jobs Detail:
  EMBED_ENTRIES: 8 failures
    Error: "Agent unreachable: embedding-worker-1"
  DISTILL_CLAIMS: 4 failures
    Error: "Timeout waiting for agent response"

Agent Status:
  embedding-worker-1:   OFFLINE (last seen: 2 hours ago)
  embedding-worker-2:   ONLINE (processing 5 jobs)
  claims-distiller-1:   ONLINE (processing 3 jobs)
  comparison-engine:   ONLINE (idle)

Recommended Actions:
  1. [Investigate Agent] - Check why embedding-worker-1 went offline
  2. [Restart Agent] - Attempt graceful restart
  3. [Failover] - Redirect jobs to embedding-worker-2
  4. [Alert Team] - Notify SRE team

[Take Action] [View Logs] [Contact Support]
```

Step 3: Review Performance Metrics Monitor key metrics over time:

```
Performance Trends (Last 7 Days)


Response Time:
  Average: 145ms      (trending up slightly)
  P99: 1,200ms
  Max: 5,432ms (Jan 29, 2 PM - database maintenance)

Error Rate:
  0.003%      (very stable, low)
  Errors: 347 (mostly timeout errors)

Job Processing Time:
  Average: 8.5 seconds per entry
  Bottleneck: Vector embeddings (7.2s per entry)
  P99: 22 seconds

Uptime:
  99.97% (2 incidents: Jan 29 maintenance, Jan 15 database failover)
  Target: 99.95%   Exceeded
```

Step 4: Set Up Alerts Configure notifications for issues:

```
[Create Alert]

Alert Name: Job Queue Backlog Critical

Trigger Condition:
  Pending jobs > 100 AND In-progress jobs < 2
  (indicates agent/worker failure)

Condition Evaluation: Every 5 minutes

Action:
    Send Slack notification to #incidents
    Email sre-team@company.com
    Create incident ticket
    Auto-restart agents (risky)

Escalation:
  If unresolved after 30 minutes, page on-call SRE

[Save Alert] [Test Alert]
```

Verification

Confirm system monitoring is effective:

- ☐ Dashboard shows current health
- ☐ Issues are detected quickly
- ☐ Alerts are configured and working
- ☐ Performance trends are visible
- ☐ You can drill down into problems
- ☐ Recommended actions are clear

Workflow 4: Agent Management

Overview

Register, configure, and manage ThinkerAgents globally. Monitor agent health and handle agent-related incidents.

Use case: You need to deploy a new embeddings agent for faster search indexing. You register it, monitor startup, and ensure it begins processing jobs.

Related workflows: - Organization Administrator — Org-level agent configuration - System Monitoring — Monitor agent health

Prerequisites

- ☐ System admin access
- ☐ Agent software deployed or ready to deploy
- ☐ Understanding of agent types and capabilities

Step-by-Step Instructions

Step 1: Access Agent Management **Navigate to:** Admin → Agents

```
Agent Management


[+ Register Agent] [Import Agents]

Active Agents: 6
```

```
Agent Fleet Status:


Name                        | Type       | Org      | Status  | Load

embedding-worker-1         | Embedding  | Global   |  Online | 4/5
embedding-worker-2         | Embedding  | Global   |  Online | 2/5
claims-distiller-1         | Claims     | Global   |  Online | 3/5
comparison-engine          | Comparison | Global   |  Online | 1/5
custom-processor-acme      | Custom     | ACME Inc |  Online | 0/3
research-embedder          | Embedding  | Research |  Starting| N/A
```

Step 2: Register New Agent Click "[+ **Register Agent**]":

```
Register New ThinkerAgent



Agent Details:
  Name *: [research-embedder]
  Type *: [Embedding  ]
  Organization *: [Research  ]

  Description:
  [High-performance embedding service for research notebooks]

Deployment:
  Infrastructure Location: [us-west-2-prod]
  Health Check Endpoint: [https://agent.research.internal:8080/health]
  Max Concurrent Jobs: [10]

Security:
  Max Classification: [CONFIDENTIAL  ]
  Compartments: [Select...]
     Research
     Executive
     Operations

Credentials:
  [Generate Credentials]

  Agent ID: (will be generated)
  Token: (will be shown once)

[Register & Generate Credentials] [Cancel]
```

Step 3: Deploy Agent After registration, get credentials:

```
 Agent Registered!

research-embedder

Agent ID: research-embedder-abc123
Token: eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...

  Deploy agent with these credentials:

Environment Variables:
  export CYBER_AGENT_ID=research-embedder-abc123
  export CYBER_AGENT_TOKEN=eyJ...
  export CYBER_SERVER=https://cyber.company.com

Deployment Steps:
  1. Copy credentials to agent's deployment environment
  2. Start agent process/container
  3. Agent will connect and report health status
  4. Status will change to "Online" when healthy

Monitor Deployment:
  [Refresh Status] [View Agent Logs]
```

Step 4: Monitor Agent Health

```
Agent: research-embedder


Status:  Starting (2 minutes since registration)
Last Heartbeat: Never (agent hasn't connected yet)
Uptime: N/A

Expected in next 5 minutes:
  • Agent connects and sends first heartbeat
  • Status changes from "Starting" to "Online"
  • Agent becomes eligible for job assignments

Actions:
  [Refresh Status] [View Deployment Logs]
  [Check Network Connectivity] [Force Health Check]

If still not online after 10 minutes:
  [Investigate] [Restart Agent] [Rollback Deployment]
```

Once online:

```
Agent: research-embedder


Status:  Online (healthy)
Last Heartbeat: 30 seconds ago
Uptime: 8 minutes

Performance:
  CPU: 45%
  Memory: 2.1 GB / 4 GB
  Jobs Processed: 12
  Failed Jobs: 0
  Average Job Time: 7.8 seconds

Current Load:
  In Progress: 2/10 jobs
  Queue Wait: 0 (processing immediately)

Recent Jobs:
  [View Last 10] [Export Job Log]

Actions:
  [Update Config] [Rotate Credentials]
  [Set Performance Limits] [Pause Agent] [Deregister]
```

Step 5: Manage Agent Fleet For multiple agents, manage them together:

```
Agent Fleet Management


Load Balancing:
  Total Jobs Pending: 47
  Distribution (auto-balancing):
    embedding-worker-1: 2/5 (40%)
    embedding-worker-2: 2/5 (40%)
    research-embedder: 1/10 (10%)

Scaling Recommendations:
    Current capacity is sufficient (70% avg utilization)
    If load increases 50%+, add another embedding agent

Alerts & Policies:
  [Max concurrent jobs per agent: 10]
  [Min agents per job type: 1] (prevent single point of failure)
  [Auto-restart on failure: Enabled]
  [Credential rotation: Every 90 days]

[Edit Policies] [Scale Fleet] [View Metrics]
```

Verification

Confirm agent management is working:

- ☐ New agent registers successfully
- ☐ Credentials are securely issued
- ☐ Agent connects and comes online
- ☐ Health checks pass
- ☐ Jobs are being assigned
- ☐ Failed jobs are handled appropriately

Summary: Quick Reference

The 4 Workflows at a Glance

Workflow	Purpose	Time	Frequency
1. User Management	Create/manage accounts	10-20 min	As needed
2. Quota Management	Set usage limits	10-15 min	Quarterly
3. System Monitoring	Health & performance	5-10 min	Daily
4. Agent Management	Deploy/manage agents	20-30 min	Quarterly

Related Personas

Your workflows overlap with:

- **Organization Administrator** — Who manage organization-level settings
- **ThinkerAgent Operator** — Who deploy agents operationally
- **Auditor** — Who review your admin actions

Last updated: February 21, 2026 **Chapter version:** 1.0.0 (Beta) **Platform version:** 2.1.0

- Deploy ThinkerAgent instances
- Configure Ollama and embedding endpoints
- Monitor worker health and performance
- Handle job failures and retries
- Optimize resource utilization
- Troubleshoot agent issues

Required Permissions: - Infrastructure access (SSH, container platforms) - Agent registration credentials - System admin or operator role

Typical Workflows: 3 core workflows in this chapter

Workflow 1: Deploying ThinkerAgents

Overview

Deploy agent instances to infrastructure, connect them to Cyber, and verify health.

Use case: You're deploying a new embeddings agent to speed up search indexing. You provision the infrastructure, start the agent, and verify it connects to Cyber.

Related workflows: - Monitoring Job Queues — Monitor jobs agent processes - Configuring Agents — Register agents in Cyber

Prerequisites

- ☐ Agent credentials from Cyber admin
- ☐ Infrastructure provisioned (VM, container, cloud instance)
- ☐ Network connectivity to Cyber server
- ☐ Ollama or embedding service running (if needed)

Step-by-Step Instructions

Step 1: Provision Infrastructure Provision a server or container for the agent:

```
# Example: Deploy as Docker container

docker run -d \
  --name cyber-embedding-worker \
  --restart unless-stopped \
  -e CYBER_AGENT_ID=research-embedder-abc123 \
  -e CYBER_AGENT_TOKEN=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9... \
  -e CYBER_SERVER=https://cyber.company.com \
  -e OLLAMA_URL=http://ollama:11434 \
  -e WORKER_THREADS=8 \
  -e MAX_QUEUE_SIZE=100 \
  -p 8080:8080 \
  cyber/embedding-worker:latest

# Verify container is running
docker ps | grep cyber-embedding-worker
```

Step 2: Configure Worker Environment Set up environment variables:

```
# Create .env file
cat > worker.env << EOF
# Cyber Connection
CYBER_AGENT_ID=research-embedder-abc123
CYBER_AGENT_TOKEN=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
CYBER_SERVER=https://cyber.company.com
CYBER_HEARTBEAT_INTERVAL=30s

# Embedding Service
OLLAMA_URL=http://ollama.internal:11434
EMBEDDING_MODEL=nomic-embed-text
EMBEDDING_DIMENSION=768

# Worker Configuration
WORKER_THREADS=8
MAX_QUEUE_SIZE=100
JOB_TIMEOUT=30m
RETRY_ATTEMPTS=3
RETRY_BACKOFF=exponential
```

```
# Monitoring
LOG_LEVEL=info
PROMETHEUS_PORT=9090
HEALTH_CHECK_PORT=8080

# Security
TLS_ENABLED=true
TLS_CERT=/etc/agent/cert.pem
TLS_KEY=/etc/agent/key.pem
EOF

# Load environment
source worker.env
```

Step 3: Start Agent Start the agent process:

```
# Start agent from binary
./cyber-worker start --config worker.env

# Or start via systemd (for persistent agents)
sudo systemctl start cyber-embedding-worker
sudo systemctl enable cyber-embedding-worker

# Check status
sudo systemctl status cyber-embedding-worker
```

Step 4: Verify Connection Check that agent connects to Cyber:

```
# Check agent logs
docker logs cyber-embedding-worker | tail -20

# Expected output:
# [INFO] Connecting to https://cyber.company.com...
# [INFO] Heartbeat sent successfully
# [INFO] Agent registered: research-embedder-abc123
# [INFO] Ready to process jobs

# Verify health endpoint
curl http://localhost:8080/health
# Expected: {"status":"healthy","uptime":"2m30s","jobs_processed":0}
```

Step 5: Verify in Cyber Admin Panel In Cyber, check agent status (Admin → Agents):

```
Agent: research-embedder
Status:   Online (healthy)
Last Heartbeat: 2 minutes ago
Uptime: 2 minutes 30 seconds
Jobs Processed: 0 (waiting for first job)

Performance:
  CPU: 5% (idle)
  Memory: 0.8 GB / 4 GB
  Network: 10 Mbps (heartbeat)
```

Verification

Confirm agent deployment is successful:

- □ Container/process is running
- □ Environment variables are set
- □ Health endpoint responds 200
- □ Agent appears in Cyber admin panel as "Online"

- □ Heartbeat is being sent regularly
- □ No errors in logs

Workflow 2: Configuring Ollama and Embeddings

Overview

Set up and configure Ollama (or other embedding service) for agents to use.

Use case: You need to switch to a faster embedding model. You update Ollama configuration, pull the new model, and restart agents.

Related workflows: - Deploying Agents — Agents depend on Ollama - Monitoring Performance — Monitor embedding latency

Prerequisites

- □ Ollama or embedding service installed
- □ GPU available (optional but recommended)
- □ Storage for models (~2-8 GB per model)

Step-by-Step Instructions

Step 1: Install Ollama

```
# macOS
brew install ollama

# Linux
curl https://ollama.ai/install.sh | sh

# Windows
# Download installer from https://ollama.ai/download

# Verify installation
ollama --version
ollama serve  # Start Ollama service (runs on port 11434)
```

Step 2: Pull Embedding Models

```
# In another terminal, pull models
ollama pull nomic-embed-text   # Recommended: fast, accurate
ollama pull mxbai-embed-large  # Alternative: higher quality
ollama pull all-minilm-l6-v2   # Legacy: lightweight

# List available models
ollama list

# Output:
# NAME                          SIZE      MODIFIED
# nomic-embed-text:latest       274 MB    2 hours ago
# mxbai-embed-large:latest      669 MB    5 hours ago
# all-minilm-l6-v2:latest       92 MB     1 day ago
```

Step 3: Configure Agent to Use Model

```
# In worker.env, specify embedding model
EMBEDDING_MODEL=nomic-embed-text

# Restart agent to pick up new model
docker restart cyber-embedding-worker

# Verify model is being used
```

```
curl http://localhost:8080/model
# {"model":"nomic-embed-text","dimension":768,"latency_ms":12}
```

Step 4: Monitor Embedding Performance

```
# Check embedding latency
curl http://localhost:8080/metrics | grep embedding_latency

# Output:
# embedding_latency_ms: 12.3 (average)
# embedding_latency_p99_ms: 45.2 (99th percentile)

# If latency is high (>100ms), consider:
# 1. Reduce WORKER_THREADS (less contention)
# 2. Add GPU (GPU_ENABLED=true in env)
# 3. Switch to faster model (nomic-embed-text is fastest)
```

Step 5: Scale Ollama (Optional) For high-load scenarios, run Ollama separately:

```
# Run Ollama on dedicated machine/container
docker run -d \
  --name ollama \
  --gpus all \
  -v ollama-data:/root/.ollama \
  -p 11434:11434 \
  ollama/ollama:latest

# Pull models into this instance
docker exec ollama ollama pull nomic-embed-text

# Configure agents to point to this Ollama instance
OLLAMA_URL=http://ollama.internal:11434
```

Verification

Confirm Ollama is working:

- ☐ Ollama service is running
- ☐ Models are pulled and available
- ☐ Agents can connect and request embeddings
- ☐ Embedding latency is acceptable (< 50ms)
- ☐ No OOM or GPU errors in logs

Workflow 3: Monitoring Worker Health

Overview

Monitor agent health, diagnose issues, and handle failures.

Use case: An embedding agent stopped processing jobs. You check its status, see it ran out of memory, restart it, and increase memory allocation.

Related workflows: - System Monitoring — Platform-wide health - Deploying Agents — Agent deployment

Prerequisites

- ☐ Agent(s) deployed and running
- ☐ Access to agent logs and metrics
- ☐ Monitoring/alerting system (optional)

Step-by-Step Instructions

Step 1: Check Agent Status

```
# Check via Cyber admin panel
# Admin → Agents → [Select agent]

# Or via API
curl -H "Authorization: Bearer TOKEN" \
  https://cyber.company.com/api/agents/research-embedder-abc123

# Response:
# {
#   "id": "research-embedder-abc123",
#   "status": "online",
#   "last_heartbeat": "2026-01-31T15:30:00Z",
#   "uptime": "2h15m",
#   "cpu_percent": 65,
#   "memory_mb": 2100,
#   "memory_limit_mb": 4096,
#   "jobs_in_progress": 3,
#   "jobs_completed": 1247,
#   "jobs_failed": 2
# }
```

Step 2: Review Agent Logs

```
# Check container logs
docker logs cyber-embedding-worker | tail -50

# Or systemd logs
journalctl -u cyber-embedding-worker -f

# Look for:
# [ERROR] Job processing failed
# [WARN] Memory usage at 85%
# [ERROR] Connection to Ollama lost
# [ERROR] Out of memory (OOM)
```

Step 3: Diagnose Common Issues Issue: Agent shows “Offline”

```
# Check if container is running
docker ps | grep cyber-embedding-worker
# If not running: docker start cyber-embedding-worker

# Check network connectivity
curl -I http://cyber.company.com
# Should return HTTP 200

# Check credentials
echo $CYBER_AGENT_TOKEN | cut -d'.' -f1  # First part of JWT
# Should be valid token format
```

Issue: High Memory Usage

```
# Check current memory
docker stats cyber-embedding-worker
# Look for memory % and actual usage

# Reduce WORKER_THREADS
# Increase memory limit
docker update --memory 8g cyber-embedding-worker
docker restart cyber-embedding-worker
```

Issue: Job Processing is Slow

```
# Check embedding latency
curl http://localhost:8080/metrics | grep latency
```

```
# Check CPU usage
docker stats --no-stream cyber-embedding-worker
# If CPU < 50%, increase WORKER_THREADS
# If CPU > 95%, add more agents or reduce threads

# Check queue depth
curl http://localhost:8080/queue
# If queue > max_size, agent is overwhelmed
```

Step 4: Restart Agent

```
# Graceful restart (finish current jobs)
docker restart cyber-embedding-worker

# Verify it reconnects
sleep 10
docker logs cyber-embedding-worker | grep "registered"
# Should see: "Agent registered: research-embedder-abc123"

# Check in Cyber admin
# Status should be "Online" within 30 seconds
```

Step 5: Set Up Monitoring

```
# Export metrics for monitoring
docker run -d \
  -p 9090:9090 \
  -v /etc/prometheus/prometheus.yml:/etc/prometheus/prometheus.yml \
  prom/prometheus:latest

# Prometheus config includes:
# - scrape_interval: 15s
# - targets: ['localhost:8080/metrics']
# - alert rules for high CPU, memory, queue depth

# Set up alerts
# If memory > 85% for 5 minutes → Page on-call
# If offline for > 2 minutes → Page on-call
# If queue depth > 500 → Alert (but don't page)
```

Verification

Confirm monitoring is effective:

- ☐ Agent status is visible
- ☐ Logs can be accessed
- ☐ Metrics are being collected
- ☐ Issues are caught quickly
- ☐ Restart procedure works
- ☐ Agents recover automatically

Summary: Quick Reference

The 3 Workflows at a Glance

Workflow	Purpose	Time	Frequency
1. Deploy Agents	Set up new workers	15-30 min	Quarterly
2. Configure Ollama	Set up embedding service	10-20 min	As needed
3. Monitor Health	Diagnose issues	5-15 min	Daily

Your Agent Operating Cycle

```
1. Deploy Agent (once)
   ↓
2. Configure Ollama (once)
   ↓
3. Monitor Health (continuous)
   ↓
4. Optimize Performance (quarterly)
   ↓
5. Scale as Needed (annually)
```

Related Personas

Your workflows overlap with:

- **System Administrator** — Register agents globally
- **Organization Administrator** — Configure agents per org
- **Notebook Owner** — Monitor jobs agents process

Last updated: February 21, 2026 **Chapter version:** 1.0.0 (Beta) **Platform version:** 2.1.0

- Create subscriptions to external notebooks
- Monitor cross-organization data flows
- Ensure Bell-LaPadula compliance (information flow rules)
- Manage inter-org security policies
- Prevent subscription cycles
- Audit cross-org access patterns

Required Permissions: - “Admin” access to your organization - Access to partner organizations (with appropriate clearance) - Understanding of Bell-LaPadula model (Chapter 2)

Typical Workflows: 3 core workflows in this chapter

Workflow 1: Setting Up Subscriptions

Overview

Create and manage subscriptions to external notebooks, configuring scope and filtering.

Use case: Your research team wants to stay informed on competitor research. You subscribe to a partner’s “Public Research” notebook and mirror entries weekly.

Related workflows: - Notebook Subscriptions — Notebook-level subscriptions - Monitoring Flows — Track synced data

Prerequisites

☐ Partner organization and notebook identified
☐ Read access to partner’s notebook
☐ Organization admin access (for org-level subscriptions)
☐ Clear purpose for subscription

Step-by-Step Instructions

Step 1: Find External Notebook **Navigate to:** Admin → Organizations → Subscriptions

```
Cross-Organization Subscriptions


Active Subscriptions (2):
  Partner A - "Public Research"   (47 entries synced)
  Partner B - "Industry Standards" (12 entries synced)

[+ Create Subscription]
```

Click “[+ **Create Subscription**]”.

Step 2: Select Source Organization

```
Create Cross-Organization Subscription


Source Organization *
[Search or select...]

Available Partner Organizations:
   ResearchCorp (10 public notebooks)
   TechPartners (5 public notebooks)
   StandardsBody (3 public notebooks)

Your Organization: MyCompany
  (Your current organization)
```

Select a partner organization.

Step 3: Select Source Notebook

```
Select Notebook from ResearchCorp


Public Notebooks (you have Read access):
   Research / AI Trends
   Research / Competitive Analysis
   Research / Public Research (currently selected)
   Standards / Industry Guidelines

Classification: PUBLIC / {}
Owner: ResearchCorp / Research Team

You can subscribe to this notebook.
```

Select the notebook.

Step 4: Configure Subscription

```
Subscription Settings


Source: ResearchCorp / Research / Public Research
Target Organization: MyCompany

Subscription Scope *
[Select what to mirror...]

   Catalog only (titles, metadata, topics)
   Catalog + Claims (above + extracted claims)
   Entries (full content, claims, metadata)

Discount Factor *
[Adjust relevance weight...]

  100% = These entries are equally relevant locally
  50%  = These entries are supplementary/reference
  10%  = These entries are minimal relevance
```

```
Polling Configuration:
  Interval: [Every 4 hours  ]
  Auto-subscribe to new entries:

Topic Filter (optional):
  [Include topics matching...]
  Examples: research/ai, research/ml
  (Leave blank to subscribe to all topics)

Information Flow Verification:


Checking Bell-LaPadula Compliance...
  Source classification: PUBLIC / {}
  Your organization min: CONFIDENTIAL / {}
   COMPLIANT (PUBLIC can flow to higher)

  Potential cycles: None detected
   NO CYCLES

[Subscribe] [Cancel]
```

Step 5: Activate Subscription

```
 Subscription Created!

ResearchCorp / Research / Public Research
→ MyCompany (org-level subscription)

Status: Syncing (initial sync in progress)
Scope: Catalog + Claims
Discount: 50%
Polling: Every 4 hours

Mirroring Progress:
  Copied: 47/47 entries
  Synced: 34/47 claims
  Status: 95% complete (ETA 5 minutes)

Next Steps:
  1. Initial sync will complete in ~5 minutes
  2. Check entries appear in your notebooks
  3. Verify access and permissions
  4. Monitor sync health

[View Progress] [Manage Subscription] [Done]
```

Verification

Confirm subscription is working:

- ☐ Subscription appears in your subscriptions list
- ☐ Initial sync completed
- ☐ Entries are visible in destination notebooks
- ☐ Sync status shows “Healthy”
- ☐ No security violations detected
- ☐ Access is restricted appropriately

Workflow 2: Monitoring Cross-Organization Flows

Overview

Track what data is being synced, monitor sync health, and investigate issues.

Use case: One of your org's subscriptions hasn't synced in 24 hours. You check the status and find the partner org's notebook was reclassified, breaking the subscription agreement.

Related workflows: - Setting Up Subscriptions — Create subscriptions - Compliance — Verify policy compliance

Prerequisites

- □ Subscriptions already created
- □ Access to subscription status dashboard
- □ Understanding of expected sync patterns

Step-by-Step Instructions

Step 1: View Subscription Dashboard **Navigate to:** Admin → Organizations → Subscriptions

```
Subscriptions Dashboard


Active Subscriptions: 3

ResearchCorp / Public Research


Status:   Healthy (last sync: 1 hour ago)
Mirrored: 47 entries (34 claims)
Watermark: Position 1,247 (all caught up)
Next sync: In 3 hours

Sync History:
  Last 7 days: 42 successful syncs, 0 failed
  Average time: 8 minutes
  Reliability: 100%

[View Mirrored Entries] [Sync Now] [Edit] [Unsubscribe]

---

TechPartners / Industry Standards


Status:    SLOW (last sync: 24 hours ago)
Mirrored: 12 entries
Watermark: Position 384 (lagging by 8 positions)
Next sync: In 2 hours (overdue)

Last Sync Error:
  "Classification changed: PUBLIC → SECRET"
  "Subscription violates information flow rule"
  "Source is now more classified than allowed"

Sync History:
  Last 7 days: 4 successful, 3 failed
  Average time: 15 minutes
  Reliability: 57%

Actions Needed:
  [Review Classification] [Contact Partner] [Pause] [Unsubscribe]
```

```
---

StandardsBody / Guidelines


Status:  Healthy
Mirrored: 89 entries (all at position 2,156)
Last sync: 4 hours ago
[Details...]
```

Step 2: Investigate Sync Failures Click on the failing subscription for details:

```
Subscription Issue: TechPartners / Industry Standards


Problem:
  Sync Status: FAILED
  Error: "Classification Conflict"
  Last Successful Sync: 24 hours ago
  Failed Attempts: 3 (automatic retries exhausted)

Root Cause:
  The source notebook classification changed:
    Was: PUBLIC / {} (allowed to sync to our org)
    Now: SECRET / {Industry} (MORE RESTRICTED)

  Bell-LaPadula Rule Violation:
    Information cannot flow DOWN in classification
    (We can't receive SECRET data in a PUBLIC subscription)

  Options:
    1. Request access to SECRET / {Industry} label
    2. Cancel subscription
    3. Wait for source to revert classification

Timeline:
  24-Jan 4:00 PM: Last successful sync (47 entries)
  25-Jan 10:30 AM: Classification changed by TechPartners
  25-Jan 10:35 AM: Sync failed (detected immediately)
  25-Jan 10:45 AM: Automatic retry failed
  25-Jan 11:00 AM: 2nd retry failed

[Contact Partner] [Review Policy] [Request Upgrade] [Cancel]
```

Step 3: Manage Watermark The watermark tracks sync progress:

```
Watermark Management


Current Watermark: Position 384
Source Notebook Position: Position 392
Behind By: 8 entries

Entries Not Yet Synced:
  Position 385: "Q1 Forecast" (created 2 hours ago)
  Position 386: "Competitor Analysis" (created 1 hour ago)
  ... (6 more entries)

When subscription is fixed:
  1. Sync will retry from position 384
  2. All 8 pending entries will be processed
  3. Watermark will advance to 392

Manual Watermark Adjustment (advanced):
  Current: 384
  New value: [____________] (careful, can skip entries)

  WARNING: Manually advancing watermark will skip entries!
  Only do this if you're certain you don't want them.

[Advance Watermark] [Reset to Last Good] [Cancel]
```

Step 4: Manually Sync if Needed Force an immediate sync:

```
[Sync Now]

Starting sync for: TechPartners / Industry Standards

Status: Attempting sync...
  • Connecting to TechPartners
  • Verifying subscription authorization
  • Checking classification compliance
  • Fetching new entries (since position 384)

Note: May still fail if underlying issue (classification conflict)
isn't resolved first.

[View Live Log] [Cancel Sync]
```

Verification

Confirm monitoring is effective:

- ☐ All subscriptions show healthy status
- ☐ Failed syncs are detected immediately
- ☐ Watermark is advancing regularly
- ☐ Sync logs are accessible
- ☐ You can manually trigger syncs
- ☐ Issues can be diagnosed

Workflow 3: Ensuring Classification Compliance

Overview

Verify that information flows comply with Bell-LaPadula rules and organizational policies.

Use case: You need to verify that all your cross-org subscriptions comply with security policy before a compliance audit.

Related workflows: - Setting Up Subscriptions — Create subscriptions - Monitoring Flows — Track syncs

Prerequisites

- ☐ Understanding of Bell-LaPadula model (Chapter 2)
- ☐ Clear organizational policy for cross-org sharing
- ☐ Access to subscription and classification data

Step-by-Step Instructions

Step 1: Review Classification Rules Verify Bell-LaPadula compliance:

```
Bell-LaPadula Compliance Check
```

```
Rule: Information flows only UPWARD in classification
      (Public → Confidential → Secret → Top Secret)

Your Organization Level: CONFIDENTIAL
  • Can subscribe to: PUBLIC or CONFIDENTIAL sources
  • Cannot subscribe to: SECRET or TOP_SECRET sources

Subscription Compliance Matrix:


Source Organization | Notebook Classification | Policy

ResearchCorp       | PUBLIC / {}            |  OK
TechPartners       | CONFIDENTIAL / {}      |  OK
StandardsBody      | PUBLIC / {}            |  OK
CompetitorA        | SECRET / {}            |  VIOLATION
GovernmentB        | TOP_SECRET / {Mil}     |  VIOLATION

Violations Found: 2
  1. CompetitorA subscription is TOO HIGH (SECRET)
     Action: [Review] [Remove Subscription] [Request Upgrade]

  2. GovernmentB subscription is TOO HIGH (TOP_SECRET)
     Action: [Review] [Remove Subscription] [Request Upgrade]

[Take Corrective Action]
```

Step 2: Document Information Flows Create a flow diagram:

```
Information Flow Documentation


MyCompany (CONFIDENTIAL)
     ↑         ↑         ↑
     |         |         |
  PUBLIC    PUBLIC    CONFIDENTIAL
     |         |         |
ResearchCorp  StandardsBody  TechPartners
 (PUBLIC)      (PUBLIC)      (CONFIDENTIAL)

 Compliant: All flows are UPWARD or SAME level
 No cycles detected
 No information downgrade risk

Export for Compliance Report:
  [Generate Diagram] [Export PDF] [Email Auditors]
```

Step 3: Audit Access Controls Verify authorized access:

```
Cross-Organization Access Audit


Question: Who in MyCompany has access to external data?

Research Team (5 people):
   Access to ResearchCorp / Public Research
   Access to StandardsBody / Guidelines
   Access to TechPartners / Industry Standards (should they?)

Executive Leadership (3 people):
   Access to all public notebooks
   Access to competitor data (appropriate restriction)
```

```
Database Team (7 people):
   Access to StandardsBody / Guidelines
   Need explicit access for TechPartners subscription

Recommendations:
  1. Grant Research Team → TechPartners / Industry Standards (Read)
  2. Document why Executive Leadership restricted from competitor data
  3. Grant Database Team → TechPartners / Industry Standards (Read)

[Implement Recommendations] [Document Decision] [Audit Log]
```

Step 4: Policy Compliance Report Generate compliance documentation:

```
Cross-Organization Subscription Compliance Report


Organization: MyCompany
Audit Date: January 31, 2026
Auditor: Alice Chen (Compliance Officer)

EXECUTIVE SUMMARY


Compliance Status:  COMPLIANT
  • 3/3 active subscriptions comply with Bell-LaPadula
  • 0 policy violations found
  • All information flows are appropriate
  • No cycles or downgrade risks detected

DETAILED FINDINGS


Subscriptions Reviewed:
  1. ResearchCorp / Public Research
     Classification: PUBLIC / {}
     Target: PUBLIC / {} (same level)
     Access: 15 users
     Compliance:  PASS

  2. TechPartners / Industry Standards
     Classification: CONFIDENTIAL / {}
     Target: CONFIDENTIAL / {} (same level)
     Access: 7 users
     Compliance:  PASS

  3. StandardsBody / Guidelines
     Classification: PUBLIC / {}
     Target: PUBLIC / {} (same level)
     Access: 45 users
     Compliance:  PASS

RECOMMENDATIONS


1. Implement quarterly compliance audits (currently ad-hoc)
2. Document business justification for each subscription
3. Set up automated Bell-LaPadula compliance alerts
4. Review access controls semi-annually

SIGN-OFF
```

```
Auditor: Alice Chen
Date: January 31, 2026
Signature: [Digital signature]

[Download PDF] [Email Stakeholders] [Archive]
```

Verification

Confirm compliance is documented:

- □ All subscriptions reviewed
- □ Information flows are compliant
- □ No cycles exist
- □ Access controls are appropriate
- □ Compliance report is generated
- □ Issues are documented

Summary: Quick Reference

The 3 Workflows at a Glance

Workflow	Purpose	Time	Frequency
1. Set Up Subscriptions	Connect to external notebooks	15-30 min	As needed
2. Monitor Flows	Track sync health	10-20 min	Weekly
3. Compliance	Verify Bell-LaPadula compliance	20-40 min	Quarterly

Key Principles

- **Information Flows Upward:** Can subscribe to less-classified data only
- **No Cycles:** Prevent circular data flow
- **Access Control:** Restrict access within org appropriately
- **Audit Trail:** Document all subscriptions and changes

Related Personas

Your workflows overlap with:

- **Organization Administrator** — Set org classification levels
- **Auditor** — Audit cross-org flows
- **Notebook Owner** — Manage individual subscriptions

Last updated: February 21, 2026 **Chapter version:** 1.0.0 (Beta) **Platform version:** 2.1.0

```
# Install from Python package
pip install notebook-client[mcp]

# Or run from source
git clone https://github.com/cyber/notebook-client
cd notebook-client
pip install -e ".[mcp]"
```

Configuration

Set environment variables:

```
export CYBER_URL="https://cyber.company.com"
export CYBER_TOKEN="your_jwt_token_here"
export CYBER_SKIP_SSL_VERIFY="false"  # Only for dev
```

Or configure via `~/.claude/claude_desktop_config.json`:

```
{
  "mcpServers": {
    "cyber": {
      "command": "python3",
      "args": ["-m", "notebook_client.mcp"],
      "env": {
        "CYBER_URL": "https://cyber.company.com",
        "CYBER_TOKEN": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9..."
      }
    }
  }
}
```

Operation Reference

WRITE - Create New Entry

Purpose: Create a new entry in a notebook

Parameters:

```
{
  "notebook_id": "nb_xyz789",
  "content": "Entry content (markdown, text, etc.)",
  "content_type": "text/markdown; charset=utf-8",
  "topic": "organization/engineering/architecture",
  "references": ["entry_abc123", "entry_def456"]
}
```

Response:

```
{
  "entry_id": "entry_new123",
  "position": 1247,
  "notebook_id": "nb_xyz789",
  "author_id": "author_hash",
  "created_at": "2026-01-31T15:30:00Z",
  "integration_cost": 2.15,
  "status": "probation"
}
```

Error Codes: - `403 Forbidden` — No write access to notebook - `404 NotFound` — Notebook doesn't exist - `400 BadRequest` — Invalid topic or references

REVISE - Update Entry

Purpose: Create a new revision of an existing entry

Parameters:

```
{
  "entry_id": "entry_abc123",
  "content": "Updated content",
  "reason": "Fixed typo and updated timeline"
}
```

Response:

```
{
  "entry_id": "entry_new456",
  "position": 1248,
  "original_entry_id": "entry_abc123",
  "reason": "Fixed typo and updated timeline"
}
```

READ - Get Entry Details

Purpose: Fetch full details of an entry

Parameters:

```
{
  "entry_id": "entry_abc123"
}
```

Response:

```
{
  "entry_id": "entry_abc123",
  "position": 1247,
  "notebook_id": "nb_xyz789",
  "content": "Full entry content",
  "content_type": "text/markdown",
  "author_id": "author_hash",
  "topic": "organization/engineering",
  "references": ["entry_def456"],
  "created_at": "2026-01-31T15:30:00Z",
  "integration_cost": 1.2,
  "status": "integrated",
  "revision_history": [
    {
      "position": 1248,
      "author_id": "author_hash2",
      "reason": "Updated timeline"
    }
  ]
}
```

BROWSE - List Entries

Purpose: List entries in a notebook with filters

Parameters:

```
{
  "notebook_id": "nb_xyz789",
  "topic": "organization/engineering",
  "status": "integrated",
  "friction_min": 0,
  "friction_max": 5,
  "limit": 50,
  "offset": 0
}
```

Response:

```
{
  "total": 247,
  "returned": 50,
  "entries": [
    {
      "entry_id": "entry_123",
      "title": "API Architecture",
      "author_id": "author_hash",
      "created_at": "2026-01-31T15:30:00Z",
      "integration_cost": 0.8,
      "status": "integrated",
      "topic": "organization/engineering/architecture",
      "preview": "The API is structured as..."
    }
  ]
}
```

SEARCH - Full-Text Search

Purpose: Search across all accessible notebooks

Parameters:

```
{
  "query": "kubernetes migration",
  "notebook_id": "nb_xyz789",
  "topic": "organization/infrastructure",
  "limit": 20
}
```

Response:

```
{
  "results": [
    {
      "entry_id": "entry_abc123",
      "title": "Kubernetes Migration Plan",
      "notebook_id": "nb_xyz789",
      "score": 0.98,
      "preview": "We are planning a phased migration to Kubernetes over 3 months...",
      "matches": [
        {
          "field": "content",
          "text": "...Kubernetes migration...",
          "offset": 145
        }
      ]
    }
  ]
}
```

OBSERVE - Track Changes

Purpose: Get entries added since a position

Parameters:

```
{
  "notebook_id": "nb_xyz789",
  "since_position": 1200
}
```

Response:

```
{
  "current_position": 1250,
  "since_position": 1200,
  "entries": [
    {
      "position": 1201,
      "entry_id": "entry_xyz",
      "title": "New Architecture Decision",
      "created_at": "2026-01-31T16:00:00Z",
      "author_id": "author_hash"
    }
  ]
}
```

SHARE - Grant Access

Purpose: Grant access to a notebook for a user/group

Parameters:

```
{
  "notebook_id": "nb_xyz789",
  "principal_id": "user_or_group_id",
  "access_tier": "read"
}
```

Access Tiers: - `existence` — Know it exists, can't read - `read` — Can read entries - `read+write` — Can read and create entries - `admin` — Full control

---

REST API Endpoints

All operations also available as REST endpoints:

```
# WRITE
POST /api/notebooks/{notebook_id}/entries
  -H "Authorization: Bearer TOKEN"
  -H "Content-Type: application/json"
  -d '{...}'

# REVISE
POST /api/entries/{entry_id}/revisions
  -H "Authorization: Bearer TOKEN"
  -d '{...}'

# READ
GET /api/entries/{entry_id}
  -H "Authorization: Bearer TOKEN"

# BROWSE
GET /api/notebooks/{notebook_id}/entries?status=integrated&limit=50
  -H "Authorization: Bearer TOKEN"

# SEARCH
GET /api/search?query=kubernetes%20migration&limit=20
  -H "Authorization: Bearer TOKEN"

# OBSERVE
GET /api/notebooks/{notebook_id}/changes?since=1200
  -H "Authorization: Bearer TOKEN"

# SHARE
POST /api/notebooks/{notebook_id}/access
  -H "Authorization: Bearer TOKEN"
  -d '{...}'
```

Authentication

Bearer Token (Recommended):

```
curl -H "Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9..." \
  https://cyber.company.com/api/notebooks
```

Session Cookie (Web Only):

```
curl -b "session=abc123..." https://cyber.company.com/api/notebooks
```

Error Responses

All errors follow this format:

```
{
  "error": "access_denied",
  "message": "User does not have read access to this notebook",
  "details": {
    "notebook_id": "nb_xyz789",
    "user_clearance": "CONFIDENTIAL / {}",
    "required_clearance": "SECRET / {Operations}"
  }
}
```

Common Error Codes: - `400 BadRequest` — Invalid parameters - `401 Unauthorized` — Missing or invalid token - `403 Forbidden` — Insufficient permissions - `404 NotFound` — Resource doesn't exist - `429 TooManyRequests` — Rate limit exceeded - `500 InternalServerError` — Server error

---

Last updated: February 21, 2026 **API Version:** 2.0 **Platform Version:** 2.1.0

Dashboard → Home page, system status Notebooks → Your notebooks Entries → Browse/search entries Explore → Topic hierarchy Search → Full-text search [Divider] Settings → Preferences, API tokens Profile → Account info, clearance Security → Keys, 2FA Audit Log → Your access history [Divider] Admin Panel → Admin-only features (if applicable)

```
### Key Pages

| Page | URL | Purpose | Access |
|------|-----|---------|--------|
| Dashboard | `/` | Overview, status | All users |
| Notebooks | `/notebooks` | Your notebooks | All users |
| Entries | `/entries` | Global entry list | All users |
| Explore | `/explore` | Topic browser | All users |
| Search | `/search` | Full-text search | All users |
| Profile | `/profile` | Account settings | All users |
| Settings | `/settings` | Preferences | All users |
| Audit Log | `/audit-log` | Your audit trail | All users |
| Admin Panel | `/admin` | User/org management | Admins only |

## Keyboard Shortcuts

| Shortcut | Action | Context |
|----------|--------|---------|
| `/` | Focus search box | Anywhere |
| `?` | Show help menu | Anywhere |
| `n` | New entry/notebook | In notebook |
| `e` | Edit/revise entry | On entry |
| `s` | Save | In edit mode |
| `Esc` | Close modal/exit edit | Modal/edit mode |
| `g d` | Go to Dashboard | Anywhere |
| `g n` | Go to Notebooks | Anywhere |
| `g e` | Go to Entries | Anywhere |
| `g s` | Go to Search | Anywhere |
| `j` | Next result | Search results |
| `k` | Previous result | Search results |

## Common UI Components
```

```
### Badges

| Badge | Meaning |
|-------|---------|
|   | Success/healthy |
|   | Warning/caution |
|   | Error/failed |
|   | In progress/pending |
|   | Locked/restricted |
|   | Starred/favorite |

### Status Indicators

| Status | Color | Meaning |
|--------|-------|---------|
| Integrated | Green | Stable, well-aligned |
| Probation | Yellow | New, still analyzing |
| Contested | Red | High friction, controversial |
| Offline | Gray | Agent not responding |
| Syncing | Blue | Data transfer in progress |

### Classification Labels
```

PUBLIC (open) CONFIDENTIAL (restricted) SECRET (very restricted) TOP_SECRET (maximum restriction)

With compartments: SECRET / {Operations, Database}

```
### Access Tiers
```

Existence (know it exists) Read (can view) Read+Write (can create/edit) Admin (full control)

```
## Filters

### Topic Filter
```

[Organization] > [Team] > [Subject] > [Subtopic]

Examples: organization/engineering/backend/database organization/operations/incidents/security

```
### Status Filter
```

All Statuses Integrated (stable entries) Probation (new entries) Contested (controversial)

```
### Friction Filter
```

All Friction Low (0-2) (well aligned) Medium (2-5) (some disagreement) High (5-10) (major disagreement)

```
### Date Range
```

Last 7 days Last 30 days Last year Custom: [From] to [To]

```
## Dialogs & Modals

### Confirmation Dialog
```

Are you sure?

This action cannot be undone.

[Confirm] [Cancel]

```
### Error Dialog
```

Error

Something went wrong: “Clearance insufficient for this resource”

[OK] [View Details]

```
### Success Dialog
```

Success

Entry created successfully!

Entry ID: entry_abc123 Position: 1,247

[View] [Create Another] [Close]

```
## Accessibility

- **Screen Reader:** Full ARIA labels on all elements
- **Keyboard Navigation:** Use Tab to navigate, Enter to activate
- **High Contrast:** Toggle in Settings → Appearance
- **Font Size:** Adjust in Settings → Appearance
- **Dark Mode:** Toggle in Settings → Appearance

---

**Last updated:** February 21, 2026
**UI Version:** 2.1.0
**Platform Version:** 2.1.0

---
```

```
    PUBLIC

    ↓ No

Is disclosure embarrassing but not damaging?
    ↓ Yes
    CONFIDENTIAL

    ↓ No

Would disclosure cause significant competitive/operational harm?
    ↓ Yes
    SECRET

    ↓ No

Would disclosure cause severe national/organizational impact?
    ↓ Yes
    TOP_SECRET
```

Clearance Dominance Examples

Valid Clearances

```
TOP_SECRET / {Medical, Ops}       dominates SECRET / {Ops}
SECRET / {A, B, C}                dominates SECRET / {A}
TOP_SECRET / {}                   dominates SECRET / {Anything}
```

Invalid Clearances

```
SECRET / {Ops}                    does NOT dominate SECRET / {Ops, Sec}
CONFIDENTIAL / {A, B}             does NOT dominate SECRET / {A}
```

```
 TOP_SECRET / {Ops}                 does NOT dominate TOP_SECRET / {Ops, Sec}
```

Information Flow Examples

Valid Flows (Information Flows Up)

```
PUBLIC notebook → CONFIDENTIAL user    OK
CONFIDENTIAL notebook → SECRET user    OK
SECRET notebook → TOP_SECRET user      OK
PUBLIC notebook → PUBLIC user          OK (same level)
```

Invalid Flows (Information Flows Down)

```
CONFIDENTIAL notebook → PUBLIC user    DENIED
SECRET notebook → CONFIDENTIAL user    DENIED
TOP_SECRET notebook → SECRET user      DENIED
```

Compartment Best Practices

Naming Convention

```
 Functional:
  - Medical Research
  - Infrastructure Operations
  - Customer Data
  - Executive

 Geographic:
  - North America
  - EMEA (Europe, Middle East, Africa)
  - Asia Pacific

 Project-Based:
  - Project Alpha
  - Project Bravo

 Vague:
  - Sensitive
  - Internal
  - Secret1, Secret2
  - TBD
```

Compartment Scope

```
Small organizations:    3-5 compartments
Medium organizations:   5-10 compartments
Large organizations:    10-20 compartments

  More than 20 compartments = management overhead
```

Access Control Matrix

User Types vs. Permissions

```
                Contributor   Manager   Owner   Admin

Read entries
Write entries
Revise entries
Grant access
Manage groups
Delete entries
Manage org
```

Compliance Checklists

Monthly Audit

- □ Review access logs for anomalies
- □ Verify clearances match roles
- □ Check for orphaned access (people who left)
- □ Verify classification labels are correct
- □ Audit cross-org subscriptions

Quarterly Review

- □ Full access control audit
- □ Compartment usage review
- □ Policy compliance check
- □ Generate compliance report
- □ Update security documentation

Annual Review

- □ Comprehensive security audit
- □ Policy effectiveness assessment
- □ Compartment consolidation
- □ Clearance recertification
- □ Threat assessment update

Security Incident Response

Potential Breach

1. Isolate affected systems immediately
2. Lock affected user account
3. Review audit logs for extent
4. Notify security team and auditors
5. Document incident with timestamps
6. Contact affected parties if appropriate
7. Revoke compromised credentials
8. Implement preventive measures

Access Control Misconfiguration

1. Identify incorrect access tier
2. Determine root cause
3. Correct the misconfiguration
4. Review for similar issues
5. Log incident
6. Document preventive measure

Classification Error

1. Identify entries with incorrect classification
2. Correct classification
3. Revoke access from unauthorized users
4. Review for similar errors
5. Update classification procedures
6. Train affected users

Last updated: February 21, 2026 **Chapter version:** 1.0.0 (Beta) **Platform Version:** 2.1.0

- `id` — Unique identifier (e.g., `nb_xyz789`)
- `name` — Display name (e.g., “Engineering Architecture”)
- `description` — Purpose and scope

- `owner_group_id` — Group that owns this notebook
- `classification` — Security label (e.g., `SECRET / {Operations}`)
- `created_at` — Timestamp of creation
- `position` — Current causal position (highest entry position)
- `retention_policy` — How long entries are kept

Access Tiers (per user/group): - `existence` — Know it exists but can't read - `read` — Can read all entries - `read+write` — Can read and create entries - `admin` — Full control including access management

Relationships: - Owns many entries - Belongs to organization - Owned by group - Has subscriptions (to other notebooks) - Has subscribers (other notebooks subscribe to it)

Entries

An entry is an immutable unit of knowledge.

Properties: - `id` — Unique identifier (e.g., `entry_abc123`) - `position` — Causal ordering (monotonic per notebook) - `notebook_id` — Which notebook contains this entry - `content` — The actual knowledge (binary blob) - `content_type` — MIME type (e.g., `text/markdown`) - `author_id` — Hash of author's public key - `signature` — Ed25519 cryptographic signature - `topic` — Hierarchical topic path (e.g., `org/engineering/backend`) - `references` — IDs of related entries (array) - `created_at` — Timestamp - `integration_cost` — Measure of coherence impact (0-10) - `status` — `probation`, `integrated`, or `contested`

Invariants: - Immutable once created (can only revise, not edit) - Cryptographically signed by author - Position never changes (causal ordering)

Revisions

A revision is a new version of an entry.

Properties: - `id` — Revision entry ID - `original_entry_id` — Entry being revised - `position` — New position (higher than original) - `reason` — Why this revision was made - `content` — Updated content - `author_id` — Who made the revision

Usage:

```
Entry v1 (position 100): "Initial architecture"
  ↓ revised
Entry v2 (position 101): "Updated with feedback"
  ↓ revised
Entry v3 (position 102): "Added performance metrics"
```

Readers see v3 by default; history shows all versions.

Causal Positions

Instead of timestamps, entries use causal positions.

Why Causal Positions? - No clock synchronization needed - Works in distributed systems - Consistent ordering across replicas - Immune to clock skew

Properties: - Monotonically increasing per notebook - Start at 1 - Never reused - Immutable once assigned

Example:

```
Notebook "Q1 Planning" positions:

Position 1: "Goals"          (created Jan 10, 9:00 AM)
Position 2: "Budget"         (created Jan 10, 10:00 AM)
Position 3: "Resources"      (created Jan 15, 2:00 PM)
Position 4: "Timeline"       (created Jan 10, 11:00 AM) ← out of order

Order of creation:    1, 2, 4, 3
```

```
Causal order:         1, 2, 3, 4 (positions determine order, not timestamps)
```

Integration Cost

Measures how well an entry aligns with existing knowledge.

Calculation: 1. Compare new entry against all existing entries (TF-IDF) 2. Form clusters of related entries 3. Compute coherence of clusters 4. Integration cost = disruption to coherence

Interpretation:

```
Cost 0-2:   Low friction, well-aligned
Cost 2-5:   Medium friction, some disagreement
Cost 5-10:  High friction, major disagreement
```

Status Evolution:

```
PROBATION      INTEGRATED       CONTESTED
(new)    →    (stable, low  ) → (stable, high)
              cost < 2          cost > 5
```

Computed by background jobs; retroactively updated when contradictions arise.

Job Queue

Background processing system.

Job Types: - `DISTILL_CLAIMS` — Extract claims from entries - `COMPARE_CLAIMS` — Compare claims between entries - `EMBED_ENTRIES` — Create vector embeddings - `CLASSIFY_ENTRIES` — Assign topics/categories

Job Lifecycle:

```
PENDING → IN_PROGRESS → COMPLETED
          ↓ (error)
        FAILED
```

Properties per Job: - `id` — Job ID - `type` — Job type - `entry_id` — Entry being processed - `status` — Current state - `started_at` — Timestamp - `completed_at` — Timestamp - `error` — Error message if failed

Retry Policy: - Automatic retries on failure - Exponential backoff - Max retries: 3 - Max retry age: 24 hours

---

Claims and Comparisons

Extracted knowledge units.

Claim:

```
Entry: "Database Indexing Strategy"

Extracted claims:
  • "PostgreSQL indexes improve query performance 50x"
  • "Compound indexes should match query patterns"
  • "Regular ANALYZE updates statistics"
```

Comparison:

```
Claim A (Entry 1): "Use Redis for caching"
Claim B (Entry 2): "Use Memcached for caching"

Comparison result: Similar (both caching solutions)
Friction: High (different approach to same problem)
Status: Contested (multiple valid approaches)
```

Audit Logs

Immutable record of all operations.

Properties: - `timestamp` — When operation occurred - `actor_id` — Who performed it - `action` — What they did (WRITE, READ, etc.) - `resource` — What was affected - `status` — Success or failure - `details` — Additional context - `signature` — Cryptographic proof

Retention: Permanent (7+ year minimum compliance)

Subscriptions

Cross-organization data mirroring.

Properties: - `source_notebook_id` — Remote notebook - `target_organization_id` — Receiving organization - `scope` — Catalog / Catalog+Claims / Entries - `discount_factor` — Relevance weight (0.1-1.0) - `polling_interval` — Sync frequency - `watermark` — Last synced position - `last_sync_time` — Timestamp

Constraints: - Source classification Target organization classification - No cycles (prevent circular data flow) - Bell-LaPadula compliance enforced

---

Organization Structure (DAG)

Directed acyclic graph of groups.

Properties: - `name` — Group name - `parent_ids` — Parent groups (can have multiple) - `child_ids` — Child groups - `classification` — Inherited + elevated - `compartments` — Inherited + supplemented

Example DAG:

```
        MyCompany
         /      \
    Engineering  Operations
      /    \
  Backend Infrastructure
```

Users can have complex memberships (in multiple groups).

Clearances

Security access specifications.

Properties: - `principal_id` — User or group ID - `level` — Classification level - `compartments` — Array of compartment names - `created_at` — When clearance was granted - `expires_at` — Optional expiration date

Dominance Check:

```python
def clearance_dominates(clearance, label):
    return (clearance.level >= label.level and
            label.compartments.issubset(clearance.compartments))
```

Last updated: February 21, 2026 **Chapter version:** 1.0.0 (Beta) **Platform Version:** 2.1.0

Check: 1. Do you have "Read" or higher access to this notebook? - Go to Settings → Profile → Notebooks 2. Does your clearance dominate the entry's classification? - Entry requires: `SECRET / {Ops}` - Your clearance: `CONFIDENTIAL / {Ops}` ← Too low!

Solution:

```
Request clearance upgrade from your organization admin
OR
Request Read access from notebook owner
```

“Clearance Insufficient for This Resource”

Cause: Classification mismatch

Solution:

```
Your clearance:     CONFIDENTIAL / {Ops}
Entry requires:     SECRET / {Ops}

1. You need SECRET level clearance (your org admin grants this)
2. Once granted, try again
3. Changes take effect within 5 minutes (or flush cache)
```

“Entry Not Found”

Cause: Entry deleted, doesn’t exist, or you lost access

Check:

```
1. Verify entry ID is correct
2. Check if entry was mirrored from external org (may have been deleted)
3. Try accessing via notebook instead of direct link
4. Check audit log for deletion event
```

“Quota Exceeded”

Cause: You hit a usage limit

Check:

Quota Type	What to Do
Notebooks exceeds limit	Request quota increase from admin
Entries per notebook	Archive old entries or split into 2 notebooks
Storage exceeds limit	Delete large entries or compress attachments
API calls per day	Reduce request frequency or batch operations

“Network Error: Connection Refused”

Cause: Can’t reach Cyber server

Check:

```
1. Is Cyber server online?
   curl https://cyber.company.com/api/health
   Should return: {"status":"ok"}

2. Is your internet working?
   ping 8.8.8.8

3. Is firewall blocking access?
   Check with IT/Network team

4. Is SSL certificate valid?
   curl -v https://cyber.company.com
   Look for SSL error messages
```

“Invalid Token”

Cause: JWT token is expired, malformed, or revoked

Check:

```
1. Verify token is complete (should be 3 parts separated by dots)
2. Check if token is expired (ask admin or regenerate)
3. Verify spelling/copying of token
4. Check if token was revoked (Settings → API Tokens)
```

Solution:

```
Generate new token:
  Settings → API Tokens → [+ Generate New Token]
  Copy entire token string
  Update environment variable or config file
```

"Job Processing Failed"

Cause: Background job error

Check:

```
1. What type of job failed?
   - EMBED_ENTRIES: Vector database issue
   - DISTILL_CLAIMS: NLP service unavailable
   - COMPARE_CLAIMS: Memory/timeout issue

2. Check if agent is running (Admin → Agents)
3. Review job logs (Admin → Jobs → [Job ID])
4. Check system status (Admin → Dashboard)
```

Solution:

```
1. Retry the job (Admin → Jobs → [Job] → Retry)
2. Wait and try again (may be temporary)
3. If persistent, contact admin or infrastructure team
```

MCP/API Issues

"Agent Not Responding"

Symptom: Claude can't access Cyber tools

Check:

```bash
# 1. Verify MCP server is running
ps aux | grep "notebook_client.mcp"

# 2. Check environment variables
echo $CYBER_URL
echo $CYBER_TOKEN

# 3. Test connection manually
curl -H "Authorization: Bearer $CYBER_TOKEN" \
  $CYBER_URL/api/health

# 4. Check Claude Desktop config
cat ~/.claude/claude_desktop_config.json
```

Solution:

```
1. Ensure notebook_client is installed:
   pip install notebook-client[mcp]

2. Verify credentials in .claude_desktop_config.json

3. Restart Claude Desktop:
   - Quit completely (Cmd+Q)
   - Wait 5 seconds
   - Reopen

4. If still failing, check logs:
   macOS: ~/Library/Logs/Claude/
```

“Python Import Error: No module named ‘notebook_client’ ”

Solution:

```
# Install/upgrade package
pip install --upgrade notebook-client[mcp]

# Verify installation
python3 -c "import notebook_client; print('OK')"

# Restart Claude Desktop
```

Performance Issues

“Searches are Slow”

Cause: Large index, slow network, or overloaded server

Solution:

```
1. Narrow search query (be more specific)
2. Filter by notebook or topic
3. Try again during off-peak hours
4. Report to admin if consistently slow
```

“Notebook Loading is Slow”

Cause: Large notebook (many entries) or slow connection

Solution:

```
1. Use filters (status, topic, date range)
2. Paginate results (load 50 instead of all)
3. Close other tabs/apps consuming bandwidth
4. Check network speed (speedtest.net)
```

“Agent Job Backlog Growing”

Cause: Agents can’t keep up with demand

Check:

```
Admin → Dashboard → Job Queue

If backlog > 100:
  1. Check if agents are online
  2. See how many jobs in progress (may be slow)
  3. Check if agent hit resource limit (CPU/memory)
```

Solution:

```
Short-term:
  1. Add more agents (request from infrastructure)
  2. Reduce new entry creation (less load)

Long-term:
  1. Optimize job processing (faster model, better hardware)
  2. Scale horizontally (more agents)
```

Subscription Issues

“Subscription Sync Failing”

Cause: Network, permission, or classification issue

Check:

```
Admin → Subscriptions → [Problem subscription]

Look for error message:
  • "Connection refused" → Source org unreachable
  • "Unauthorized" → Lost access to source notebook
  • "Classification changed" → Bell-LaPadula violation
  • "Notebook deleted" → Source notebook no longer exists
```

Solution:

```
If "Classification changed":
  • Request clearance upgrade if needed
  • Or unsubscribe and resubscribe

If "Connection refused":
  • Check network connectivity
  • Verify source org is online

If "Unauthorized":
  • Request Read access from source notebook owner
  • Or generate new token
```

“Entries Not Syncing”

Cause: Watermark stuck or sync paused

Check:

```
Watermark: Position 384
Source position: Position 392

Behind by 8 entries? Try:
  1. Click [Sync Now] to force immediate sync
  2. Wait 5 minutes
  3. Check subscription status for errors
```

Account Issues

“Can’t Log In”

Cause: Wrong password, account locked, or system issue

Solution:

```
1. Verify you're using correct email
2. Try password reset (Settings → Account → Reset Password)
3. If account is locked, contact admin
4. If password reset doesn't work, contact support
```

“Lost API Token”

Cause: Token wasn’t saved

Solution:

```
Generate a new token:
  Settings → API Tokens → [+ Generate New Token]

Save it securely:
  • Environment variable: export CYBER_TOKEN="..."
  • Password manager: Save the token
  • .env file: Add to git .gitignore

DO NOT:
  • Commit token to code
  • Send token in messages/email
  • Share token with others
```

Getting Help

Where to Find Help

Issue	Resource
How do I do X?	This manual + [Chapter relevant to your role]
API error	Chapter 11: MCP Integration Reference
UI question	Chapter 12: UI Reference
Security question	Chapter 2: Security Model
Configuration issue	This chapter (Troubleshooting)
Still stuck	Contact your Cyber admin or support@cyber.internal

Providing Information When Reporting Issues

```
When reporting a bug, include:
  1. What you were trying to do
  2. What actually happened
  3. Error message (if any)
  4. Steps to reproduce
  5. Your browser/version (if UI issue)
  6. Your clearance level (Settings → Profile)
  7. Relevant entry/notebook IDs
  8. Timestamp of issue occurrence

Example:
  "I tried to create an entry in my notebook at 2:30 PM today.
   I got error: 'Clearance insufficient for this resource'.
   My clearance is CONFIDENTIAL / {Ops}.
   The notebook is classified SECRET / {Ops}.
   Notebook ID: nb_xyz789"
```

Last updated: February 21, 2026 **Chapter version:** 1.0.0 (Beta) **Platform Version:** 2.1.0

A monotonically increasing sequence number that establishes the order of events in a notebook without relying on timestamps or synchronized clocks.

Cyber A multi-organization classified knowledge exchange platform with enterprise-grade security, entropy-based knowledge integration, and federated identity.

Entry An immutable unit of knowledge—a single piece of information in a notebook, characterized by content, authorship, classification, and references.

Integration Cost A numerical measure (0-10) of how well an entry aligns with existing knowledge in a notebook, based on TF-IDF similarity and coherence analysis.

Notebook A domain-specific, security-labeled knowledge space that contains entries and is managed by an owning group.

Revision A new version of an existing entry, created when information needs to be updated. The original entry remains immutable; the revision supersedes it.

Security

Access Tier A permission level controlling what operations a principal (user/group) can perform: Existence, Read, Read+Write, Admin.

Bell-LaPadula Model A formal security framework that enforces: (1) Information can only flow upward in classification, and (2) Users can only read information they're cleared for.

Classification Level A five-level hierarchy (PUBLIC, CONFIDENTIAL, SECRET, TOP_SECRET, Custom) indicating information sensitivity and distribution restrictions.

Clearance A security credential specifying what classified information a principal is authorized to access, consisting of a level + compartments.

Compartment An optional security category (e.g., "Medical Research", "Strategic Planning") that further restricts access within a classification level.

Dominance In Bell-LaPadula terms, one clearance dominates another if it has a higher or equal level AND includes all required compartments.

Information Flow The movement of data through the system. Bell-LaPadula enforces that information flows only from lower to higher classification.

Security Label A combination of classification level and compartments (e.g., `SECRET / {Operations, Database}`).

Organizational

Cross-Organization Coordinator A persona who manages knowledge sharing between organizations and ensures compliance with security boundaries.

DAG (Directed Acyclic Graph) A structure describing organizational groups where a group can have multiple parents but no cycles (e.g., Engineering / Backend).

Federated Identity A decentralized identity system using cryptographic keys (Ed25519) where users are identified by their public key hash, not usernames.

Group An organizational unit that contains users and owns notebooks. Groups form a DAG hierarchy with inherited classification.

Knowledge Contributor A persona focused on creating, discovering, and refining entries in notebooks.

Notebook Owner A persona who creates and manages notebooks, controls access, reviews submissions, and monitors processing.

Organization A top-level container representing a company or entity with its own security boundaries and group hierarchies.

System Administrator A persona managing platform-wide settings: user accounts, quotas, agents, and system health.

Technical

Audit Log An immutable record of all operations, including actor, action, resource, timestamp, and cryptographic signature.

Batch Entry Creation UI feature for importing multiple entries at once via CSV or text format.

Claim An extracted piece of knowledge from an entry, identified by NLP processing (e.g., "Database indexing improves performance 50x").

Comparison Analysis of semantic similarity between two claims or entries, used to identify disagreements or redundancy.

Embedding A vector representation of text, created by AI models, used for semantic search and similarity analysis.

Job A background processing task (DISTILL_CLAIMS, COMPARE_CLAIMS, EMBED_ENTRIES) run by ThinkerAgents.

MCP (Model Context Protocol) A protocol enabling AI systems like Claude to interact with Cyber programmatically.

Ollama An embedding service that runs AI models locally for creating text embeddings.

ThinkerAgent An AI processing worker that analyzes notebook entries and extracts claims, embeddings, and comparisons.

Watermark A tracking mechanism showing the last successfully synced position in a subscription.

Operational

Auditor/Compliance Officer A persona responsible for ensuring Cyber usage complies with security policies and investigating incidents.

Chaos Engineering Intentional disruption testing to ensure system resilience.

Coherence A measure of how consistently related entries align in meaning and approach.

Friction Another term for integration cost; high friction indicates controversial or novel entries.

Least Privilege A security principle: grant only the minimum permissions necessary for a user to do their job.

Probation An entry status indicating it's new and still undergoing integration cost analysis.

Contested An entry status indicating it has high integration cost and contradicts existing knowledge.

Integrated An entry status indicating it's stable and well-aligned with existing knowledge.

ThinkerAgent Operator A persona who deploys, configures, and monitors AI processing workers.

Index of Workflows

Workflow ID	Title	Persona	Chapter
WF-KC-001	Setting up MCP Access for Claude Desktop	Knowledge Contributor	4
WF-KC-002	Creating and Organizing Entries	Knowledge Contributor	4
WF-KC-003	Browsing and Discovering Knowledge	Knowledge Contributor	4
WF-KC-004	Searching Across Notebooks	Knowledge Contributor	4
WF-KC-005	Managing Revisions	Knowledge Contributor	4
WF-KC-006	Observing Changes	Knowledge Contributor	4
WF-OA-001	Creating Organizational Structure	Org Administrator	5
WF-OA-002	Managing Group Memberships	Org Administrator	5
WF-OA-003	Managing Security Clearances	Org Administrator	5
WF-OA-004	Configuring ThinkerAgents	Org Administrator	5
WF-NO-001	Creating and Configuring Notebooks	Notebook Owner	6
WF-NO-002	Managing Access Control	Notebook Owner	6
WF-NO-003	Reviewing Submissions	Notebook Owner	6
WF-NO-004	Monitoring Job Pipeline	Notebook Owner	6
WF-NO-005	Managing Subscriptions	Notebook Owner	6
WF-AU-001	Querying Global Audit Logs	Auditor	7

Workflow ID	Title	Persona	Chapter
WF-AU-002	Investigating Security Events	Auditor	7
WF-AU-003	Notebook-Scoped Auditing	Auditor	7
WF-SA-001	User Management	System Admin	8
WF-SA-002	Quota Management	System Admin	8
WF-SA-003	System Monitoring	System Admin	8
WF-SA-004	Agent Management	System Admin	8
WF-TO-001	Deploying ThinkerAgents	ThinkerAgent Operator	9
WF-TO-002	Configuring Ollama	ThinkerAgent Operator	9
WF-TO-003	Monitoring Worker Health	ThinkerAgent Operator	9
WF-CO-001	Setting Up Subscriptions	Cross-Org Coordinator	10
WF-CO-002	Monitoring Cross-Organization Flows	Cross-Org Coordinator	10
WF-CO-003	Ensuring Classification Compliance	Cross-Org Coordinator	10

Chapter Overview

Chapter	Title	Type	Focus
1	Platform Overview	Introduction	What Cyber is, why it exists, core concepts
2	Security Model	Introduction	Bell-LaPadula, classification, clearances
3	Getting Started	Introduction	First login, account setup, interface
4	Knowledge Contributor	Persona	Creating, discovering, managing entries
5	Organization Administrator	Persona	Structure, clearances, groups, agents
6	Notebook Owner	Persona	Creating, managing, reviewing notebooks
7	Auditor/Compliance Officer	Persona	Audit logs, investigations, compliance
8	System Administrator	Persona	Users, quotas, health, agents
9	ThinkerAgent Operator	Persona	Deployment, Ollama, monitoring
10	Cross-Organization Coordinator	Persona	Subscriptions, flows, compliance
11	MCP Integration Reference	Reference	API operations, authentication, errors
12	UI Reference	Reference	Navigation, shortcuts, components
13	Security Reference	Reference	Decision trees, examples, compliance

Chapter	Title	Type	Focus
14	Data Model	Reference	Notebooks, entries, jobs, subscriptions
15	Troubleshooting	Reference	Common errors, solutions, support
16	Glossary & Index	Reference	Terms, acronyms, workflow index

---

Acronyms

Acronym	Meaning
ACL	Access Control List
API	Application Programming Interface
CSV	Comma-Separated Values
DAG	Directed Acyclic Graph
JWT	JSON Web Token
MCP	Model Context Protocol
NLP	Natural Language Processing
OOM	Out of Memory
RBAC	Role-Based Access Control
SSH	Secure Shell
SSL/TLS	Secure Sockets Layer / Transport Layer Security
TF-IDF	Term Frequency - Inverse Document Frequency
UI	User Interface
VM	Virtual Machine
VPN	Virtual Private Network

Related Reading

For more information on security models and knowledge systems:

- **Bell and LaPadula (1973):** Original Bell-LaPadula model paper
- **NIST SP 800-95:** Guide to Secure Web Services
- **OWASP Top 10:** Common security vulnerabilities
- **Okapi BM25:** Probabilistic relevance ranking
- **Word2Vec / Embeddings:** Text representation in ML

Quick Reference: Who Does What

Task	Persona	Chapter
Create entry	Knowledge Contributor	4
Search for entry	Knowledge Contributor	4
Update entry	Knowledge Contributor	4
Create notebook	Notebook Owner	6
Grant notebook access	Notebook Owner	6
Review submissions	Notebook Owner	6
Create org structure	Org Administrator	5
Manage clearances	Org Administrator	5
Add users to groups	Org Administrator	5
Register agents	Org Administrator / System Admin	5, 8
Audit access	Auditor	7
Create users	System Administrator	8
Set quotas	System Administrator	8

Task	Persona	Chapter
Monitor system	System Administrator	8
Deploy agents	ThinkerAgent Operator	9
Monitor jobs	Notebook Owner / Operator	6, 9
Set up subscriptions	Notebook Owner / Cross-Org Coordinator	6, 10
Monitor subscriptions	Cross-Org Coordinator	10

Last updated: February 21, 2026 **Manual version:** 1.0.0 (Beta) **Platform version:** 2.1.0

Total words: ~30,000 **Total chapters:** 16 **Total workflows:** 28 **Total pages (estimated PDF):** 250+
