muni

관리자 가이드

설치·설정·계정·운영·장애 대응

버전	v0.37.0
작성일	2026-09-12
문서	관리자 가이드

muni 관리자 가이드

muni 를 설치하고, 설정하고, 계정을 만들고, 사고가 났을 때 확인하는 사람을 위한 안내입니다. 화면을 **쓰는 법**은 사용자 가이드에 있고, 백업·복구·키 교체·퇴사자 정리 같은 운영 절차의 상세는 운영 안내에 있습니다. 이 문서는 그 둘을 잇는 자리입니다.

화면은 muni v0.36.0 을 데모 데이터로 채워 실제로 찍은 것입니다(방문 추적 탭만 그 기능이 들어간 다음 빌드에서 찍었습니다).

1. 구성 요소

muni 는 **컨테이너 하나**입니다. Go 바이너리 안에 React 번들이 들어 있고, PDF 를 그릴 headless Chromium 과 Noto CJK 글꼴이 이미지에 함께 들어 있습니다. 런타임에 Redis·오브젝트 스토리지·검색 엔진이 필요하지 않습니다.

구성 요소	필수	무엇을 하나
`muni` 컨테이너	O	웹 UI, REST API(`/api/v1`), MCP(`/mcp`), 공동편집 WebSocket, Import·Export
PostgreSQL 15 이상	O	문서·버전·댓글·첨부·설정·감사 로그. **첨부까지 데이터베이스 안에 있습니다**
Keycloak	-	OIDC SSO. 붙이지 않으면 로컬 로그인만 씁니다
OpenAI 호환 게이트웨이	-	AI 기능. 끄면 AI 패널이 비활성으로 보입니다
SMTP 서버	-	계정 안내·알림 메일
Ptium	-	발표자료 생성 연동

주고받는 것은 이렇습니다. 브라우저 ↔ muni 는 HTTP 와 WebSocket(`/api/v1/collab/{id}`), muni ↔ PostgreSQL 은 DSN 한 줄, 나머지 연동은 모두 **서비스 관리 → 서비스 설정**에 저장된 주소로 muni 가 나가는 방향입니다. muni 가 밖에서 들어오는 연결을 받는 포트는 `8080` 하나뿐입니다.

2. 설치

릴리스 자산은 이미지 하나를 담은 tar.gz 입니다. 파일명과 이미지 태그가 같은 규칙을 씁니다.

```
asset: muni-v0.36.0.tar.gz
image: muni:v0.36.0
```

준비

- PostgreSQL 15 이상. DSN 계정에는 최초 실행 때 schema 와 `pgcrypto` , `citext` extension 을 만들 권한이 필요합니다.

- 디스크: 데이터베이스에 첨부까지 들어가므로 문서량이 아니라 **첨부량**으로 잡습니다.
- TLS 는 사내 ingress·reverse proxy 에서 끝내는 구성을 권장합니다.

올리기

```
gzip -dc muni-v0.36.0.tar.gz | docker load
cp .env.example .env
# .env 의 네 값을 안전한 값으로 바꿉니다
openssl rand -base64 32          # ENCRYPTION_KEY 에 넣을 값
docker compose -f compose.example.yaml --env-file .env up -d
```

`compose.example.yaml` 의 `image:` 태그를 받은 버전(`muni:v0.36.0`)으로 맞춘 뒤 올립니다. 기동하면 마이그레이션이 자동으로 적용되고, `BOOTSTRAP_ADMIN` 으로 최초 관리자 계정이 만들어 집니다. 계정이 이미 있으면 bootstrap 값은 그 계정을 덮어쓰지 않습니다.

```
curl -fsS http://127.0.0.1:8080/healthz    # 살아 있는가
curl -fsS http://127.0.0.1:8080/readyz     # 요청을 받을 준비가 되었는가
docker compose logs -f muni
```

브라우저로 `http://<host>:8080/login` 에 들어가 `BOOTSTRAP_ADMIN` 계정으로 로그인하면 끝입니다. 첫 로그인 뒤 **개인 설정 → 프로필**에서 비밀번호를 바꾸세요.

자원과 경로

항목	값
포트	`8080` (HTTP 하나. 컨테이너가 여는 유일한 포트)
볼륨	없음 — 상태는 전부 PostgreSQL 에 있습니다
tmpfs	`/tmp` 512MB. PDF 렌더링이 여기에 임시 디렉터리와 Chromium `HOME` 을 만듭니다
파일 시스템	`read_only: true` 로 돌립니다. 위 tmpfs 설정은 유지하세요
사용자	비 root(uid/gid 65532), 모든 capability drop
헬스체크	`curl --fail http://127.0.0.1:8080/healthz` (이미지에 내장)
권장 자원	request 100m CPU / 192MiB, limit 2 CPU / 2GiB (`deploy/kubernetes/muni.yaml` 기준)

Kubernetes 로 올린다면 `deploy/kubernetes/muni.yaml` 이 위 설정을 그대로 담은 예제입니다. Chromium 한 프로세스가 수백 MB 를 쓰므로 메모리 한도는 `MUNI_PDF_CONCURRENCY` 와 함께 잡으세요.

3. 설정

환경 변수

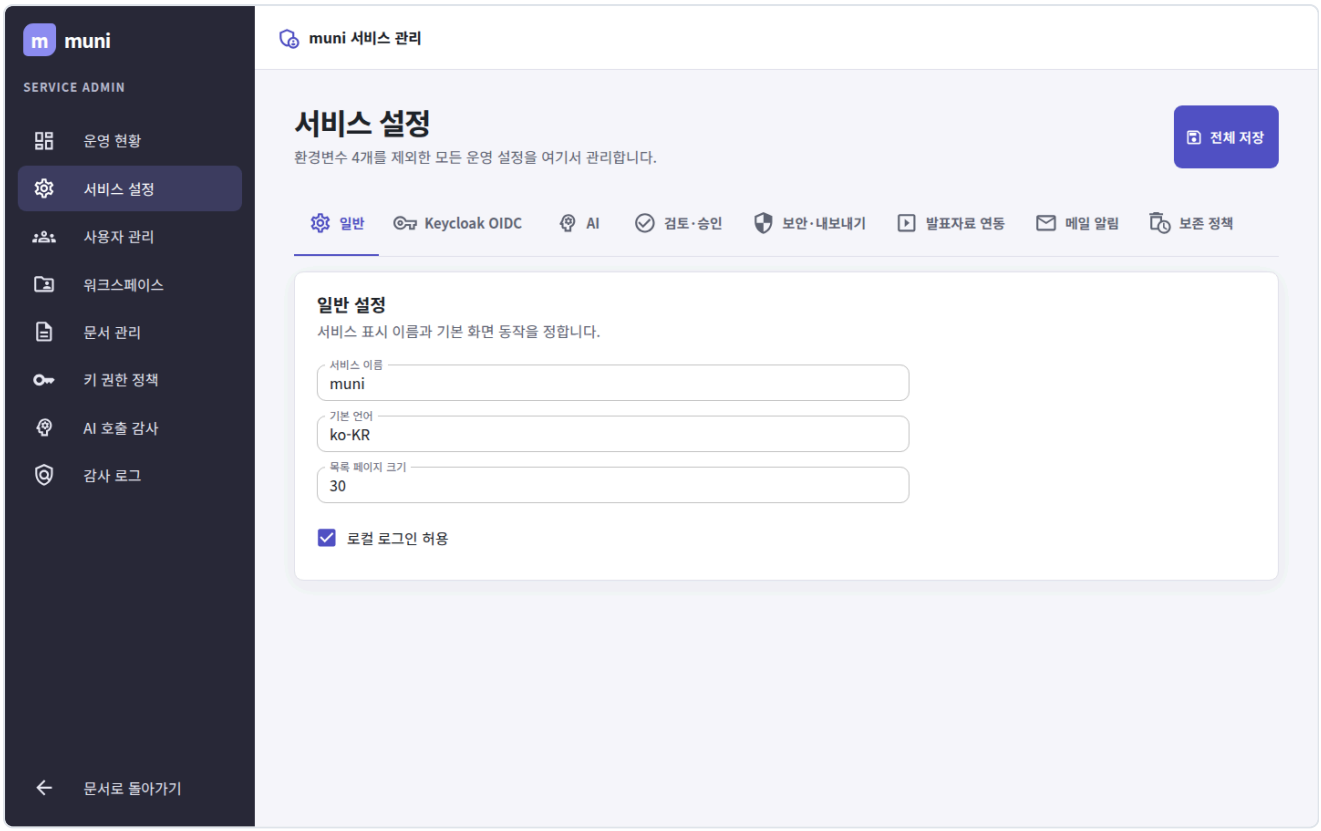
애플리케이션이 환경 변수로 받는 값은 **아래가 전부**입니다. 나머지 운영 설정은 모두 화면에서 바꿉니다.

변수	필수	기본값	설명
`POSTGRES_DSN`	O	없음	PostgreSQL 접속 문자열. 예: `postgres://muni:...@postgres.internal:5432/muni?sslmode=require`
`BOOTSTRAP_ADMIN`	O	없음	최초 관리자의 이메일 또는 3자 이상 아이디. 예: `admin@example.com`
`BOOTSTRAP_ADMIN_PASSWORD`	O	없음	최초 관리자 비밀번호. **12자 이상**. 예: `replace-with-at-least-12-characters`
`ENCRYPTION_KEY`	O	없음	설정 secret 과 사용자 data key 를 봉인하는 base64 32바이트 master key. `openssl rand -base64 32` 로 만듭니다
`MUNI_CHROMIUM_PATH`	-	자동 탐색	PDF Export 에 쓸 브라우저 실행 파일. 비우면 `chromium` · `chromium-browser` · `google-chrome` · `google-chrome-stable` · `chrome` 순으로 찾습니다
`MUNI_PDF_CONCURRENCY`	-	`2`	동시에 띄울 Chromium 수(1~32). 자리를 60초 넘게 기다리면 재시도 안내와 함께 거절합니다

값이 빠지면 기동하지 않고 `required environment variables are missing: ...` 을 남기고 멈춥니다. `ENCRYPTION_KEY` 가 base64 32바이트가 아니면 `ENCRYPTION_KEY must be a base64-encoded 32-byte key` 로 멈춥니다.

> `ENCRYPTION_KEY` 를 잃으면 봉인된 비밀값은 복구할 수 없습니다. **데이터베이스 백업과 다른 곳에** 보관하세요. 잃었을 때 할 수 있는 일은 운영 안내에 적혀 있습니다.

서비스 설정 화면



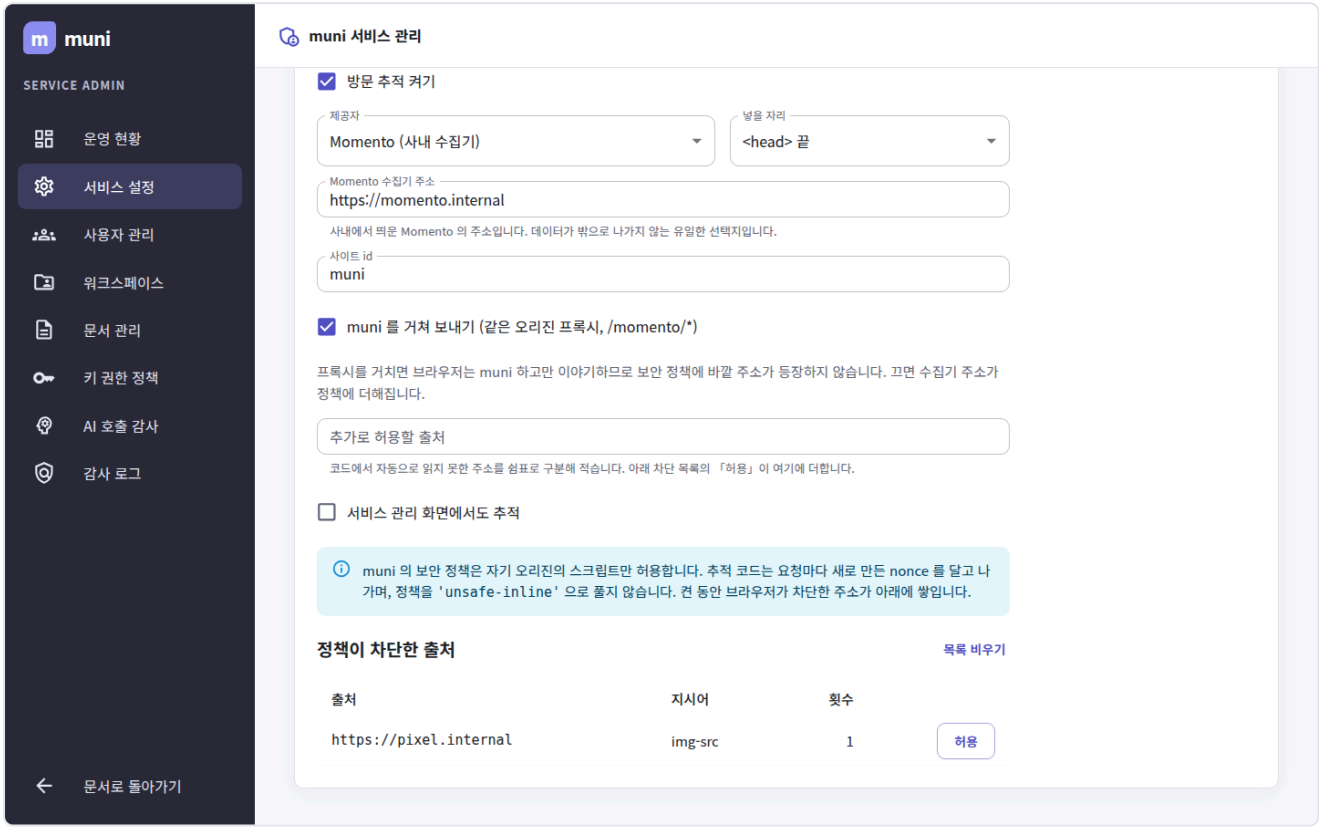
서비스 설정 — 일반·Keycloak OIDC·AI·검토·승인·보안·내보내기·발표자료 연동·메일 알림·보존 정책 탭

탭	여기서 정하는 것
일반	서비스 표시 이름, 기본 언어, 목록 페이지 크기, 로컬 로그인 허용 여부
Keycloak OIDC	issuer URL·client ID·client secret, 자동 프로비저닝과 기본 역할. Discovery 연결 테스트가 있습니다
AI	OpenAI 호환 base URL(`/v1` 까지)·API key·model·timeout·최대 토큰. 상한은 시스템 상한 `262144` 와 관리자 상한 중 작은 값입니다
검토·승인	결재 흐름 사용 여부, 필요한 승인 수, 본인 승인 허용 여부. 꺼 두면 사용자 메뉴에 「검토 및 승인」이 나타나지 않습니다
보안·내보내기	세션 유지 시간, API 키 최대 수명, 공개 링크 허용, 업로드 크기 상한(1~1024MB), 읽기 감사 로그, PDF·DOCX 내보내기 허용
발표자료 연동	Ptium 주소와 API key. 저장 전에 연결 테스트를 할 수 있습니다
메일 알림	SMTP 호스트·계정·발신 주소. 연결 테스트가 있습니다
보존 정책	휴지통·버전·감사 로그·AI 감사 로그를 며칠 뒤 지울지. 버전은 최소 5개를 남깁니다
방문 추적	어떤 화면이 쓰이는지 세는 스크립트를 페이지에 붙일지, 어느 수집기로 보낼지. 기본은 꺼짐입니다. 아래 에 따로 적었습니다

비밀값(AI API key, OIDC client secret, SMTP 비밀번호, Ptium API key)은 `ENCRYPTION_KEY` 로 봉인해 저장하며 화면에 다시 보여 주지 않습니다.

Keycloak 연동 순서는 confidential client 를 만들고 Redirect URI 에 `https://<muni-host>/api/v1/auth/oidc/callback` 을 등록한 뒤, 이 탭에 issuer·client ID·client secret 을 넣고 연결 테스트 → 활성화 → 저장입니다. Endpoint 는 discovery 로 결정하고 scope 기본 값은 `openid profile email` 입니다.

방문 추적



서비스 설정 → 방문 추적 — Momento 수집기 설정과, 정책이 차단한 출처 목록의 「허용」 단추

어떤 화면이 실제로 쓰이는지 세려면 방문 추적 스크립트를 페이지에 붙입니다. **기본은 꺼짐**이라 새로 설치한 곳에서는 아무것도 달라지지 않고, 켜더라도 muni 가 보내는 것은 스크립트가 세는 쪽 번호와 경로뿐이며 문서 내용은 담기지 않습니다. 설정은 저장소에 두고 화면에서 바꾸므로 수집기 주소가 바뀌어도 다시 배포하지 않습니다.

설정	뜻
방문 추적 켜기	꺼짐이 기본값입니다. 켜야 붙습니다
제공자	`Momento` · `Matomo` · `Google Analytics 4` · `Google Tag Manager` · `직접 붙여넣기` · `없음`
Momento 수집기 주소 · 사이트 id	사내에 띄운 Momento 수집기의 주소와 사이트 id
muni 를 거쳐 보내기	Momento 만. muni 가 `/momento/*` 를 수집기로 넘기고 스니펫은 `data-endpoint="/momento"` 로 나갑니다. **기본으로 켜져 있습니다**
측정 ID · Matomo 주소 · 사이트 id	GA4·GTM 과 Matomo 의 것. 구글 쪽은 폐쇄망에서 닿지 않습니다
추적 코드	직접 붙여넣기. **8KB** 를 넘으면 저장되지 않습니다
추가로 허용할 출처	코드에서 자동으로 읽지 못한 주소를 관리자가 더하는 자리. 아래 차단 목록의 「허용」이 여기에 더합니다
서비스 관리 화면에서도 추적	기본은 아니오. 관리자가 누른 것은 대개 알고 싶은 방문이 아닙니다
넣을 자리	`<head>` 끝 또는 `<body>` 끝

Momento 를 먼저 고르세요. Momento 는 사내 자체 호스팅 수집기라 데이터가 밖으로 나가지 않는 유일한 선택지입니다. 「muni 를 거쳐 보내기」를 켜 두면(기본) 브라우저는 muni 하고만 이야기하고 muni 가 수집기로 넘기므로 아래의 보안 정책에 바깥 주소가 아예 등장하지 않습니다. 넘길 때 방문자의 세션 쿠키는 떼고 방문자 주소는 `X-Forwarded-For` 로 붙입니다. 프록시는 Momento 를 고르고 켜 둔 동안에만 답하며, 다른 어디로도 중계하지 않습니다.

콘텐츠 보안 정책(CSP)과 왜 스니펫을 그냥 붙일 수 없는가

muni 의 모든 페이지는 `script-src 'self'` 로 잠긴 콘텐츠 보안 정책을 달고 나갑니다. 자기 오리진의 스크립트만 실행하므로 추적 코드를 그냥 끼워 넣으면 브라우저가 **조용히** 막고, 관리자는 수집기가 비어 있는 이유를 알 길이 없습니다. muni 는 그래서 다음을 함께 합니다.

1. **요청마다 nonce** — 페이지를 낼 때마다 새 nonce 를 만들어 스니펫의 모든 `<script>` 태그에 붙이고 같은 값을 정책의 `script-src` 에 넣습니다. 정책을 `'unsafe-inline'` 으로 풀지 않습니다 — 한 번 풀면 앱의 모든 인라인 스크립트가 함께 허용되고, 추적을 끈 뒤에도 느슨한 채 남기 때문입니다.
2. **출처는 코드에서 읽습니다** — 추적 도구는 자기 주소를 로더 안에 적어 둡니다. 붙여 넣은 코드에서 `http(s)` 주소를 긁어 `script-src` · `connect-src` · `img-src` 에 더합니다. Momento(직접 연결)·Matomo 는 적어 둔 주소, GA4·GTM 은 구글의 알려진 주소가 더해집니다.
3. **차단된 것을 기록합니다** — 추적이 켜진 동안에만 정책에 `report-uri /api/v1/tracking/csp-report` 를 넣습니다. 브라우저가 막은 것을 신고하면 출처와 지시어를 메모리에 기억하고(서로 다른 출처 100개까지, 데이터베이스에는 쓰지 않습니다) 이 탭의 「정책이 차단한 출처」에 보여 줍니다. 「허용」을 누르면 그 출처 하나가 「추가로 허용할

출처」에 더해지고, 화면을 새로 고치면 정책에 들어갑니다. 「목록 비우기」로 지운 뒤 다시 새로 고쳐 아직 막히는 것이 있는지 봅니다.

추적을 끄면 정책은 예전 그대로 좁아집니다. `/api/*` · `/mcp` · `/healthz` · `/readyz` · `/metrics` · `/momento/*` 같은 화면이 아닌 경로에는 스니펫이 붙지 않고, 정책은 오히려 더 좁은 `default-src 'none'` 입니다. 「서비스 관리 화면에서도 추적」이 꺼져 있으면 `/admin` 아래를 **직접 열었을 때** 스니펫이 빠지지만, 다른 화면에서 메뉴로 넘어온 경우에는 이미 실린 스크립트가 그대로 남습니다 — 화면 전환이 페이지를 다시 받지 않기 때문입니다.

설정 순서는 제공자와 주소를 넣고 「전체 저장」 → 아무 문서 화면을 새로 고침 → 수집기에서 방문이 들어오는지 확인 → 들어오지 않으면 이 탭의 차단 목록을 보고 「허용」 → 다시 새로 고침입니다.

4. 계정과 권한

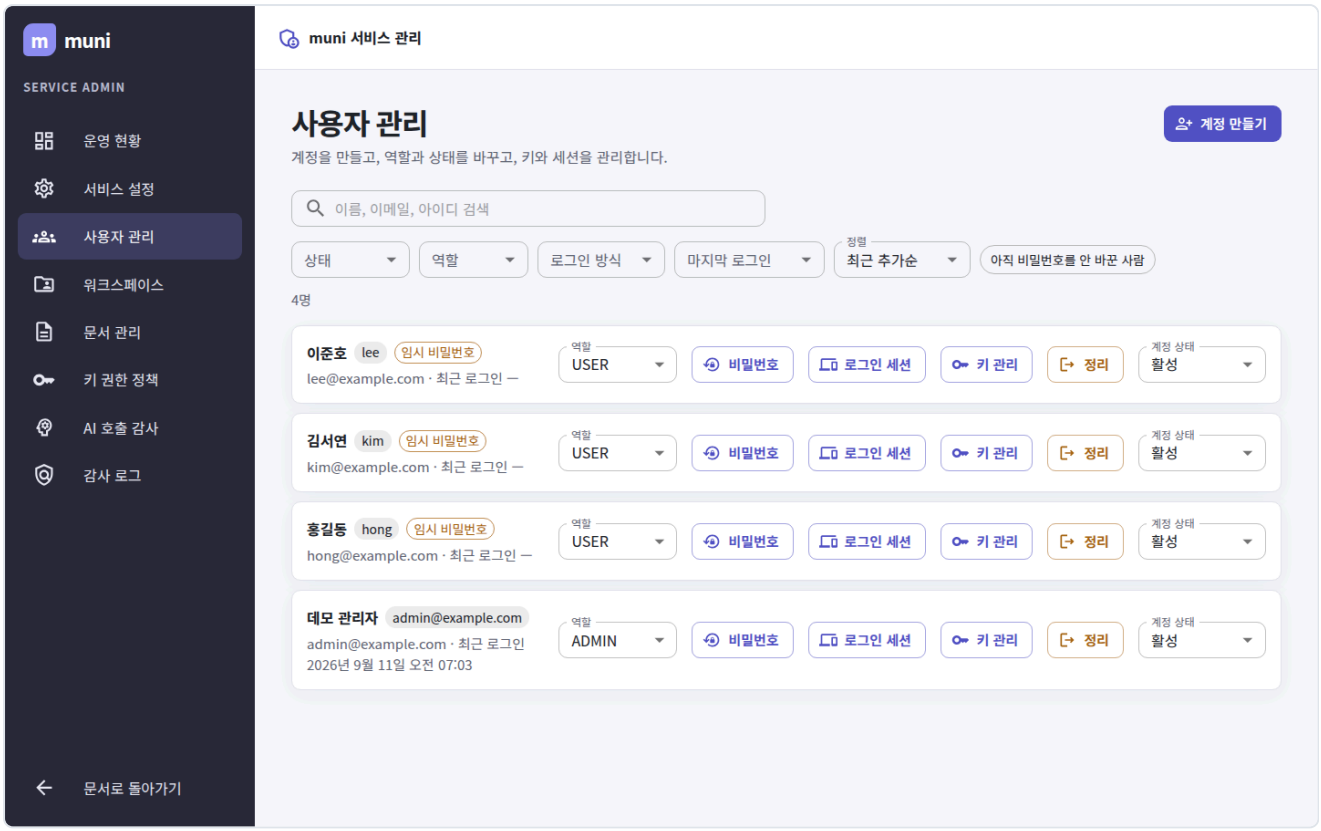
역할

muni 의 권한은 세 층입니다.

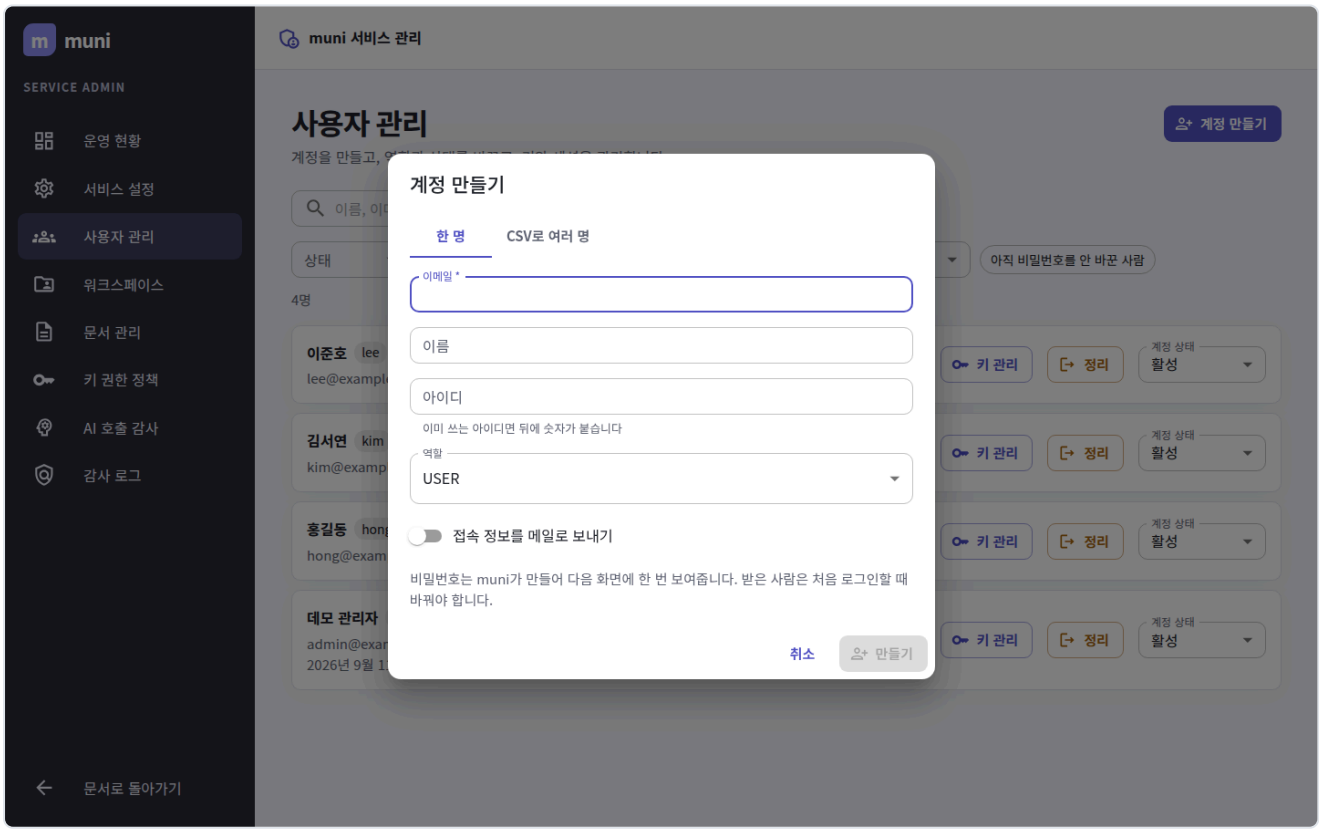
층	값	뜻
서비스 역할	`ADMIN` / `USER`	`ADMIN` 만 「서비스 관리」에 들어갑니다
워크스페이스 역할	`OWNER` / `MANAGER` / `MEMBER` / `VIEWER`	구성원 관리는 `OWNER` 와 `MANAGER` (그리고 서비스 `ADMIN`)가 합니다. `MANAGER` 는 다른 사람을 `MANAGER` 로 올릴 수 없습니다
문서 ACL	`OWNER` / `EDITOR` / `COMMENTER` / `VIEWER`	문서마다 사람에게 주는 권한. 조회·검색·AI·MCP·공동편집에 똑같이 적용됩니다

서비스 `ADMIN` 이라고 해서 남의 문서 본문이 저절로 보이지는 않습니다. 관리 화면에서 하는 일은 **소유권 이전**과 **권한 보기**이고, 그 동작은 감사 로그에 남습니다.

계정 만들기



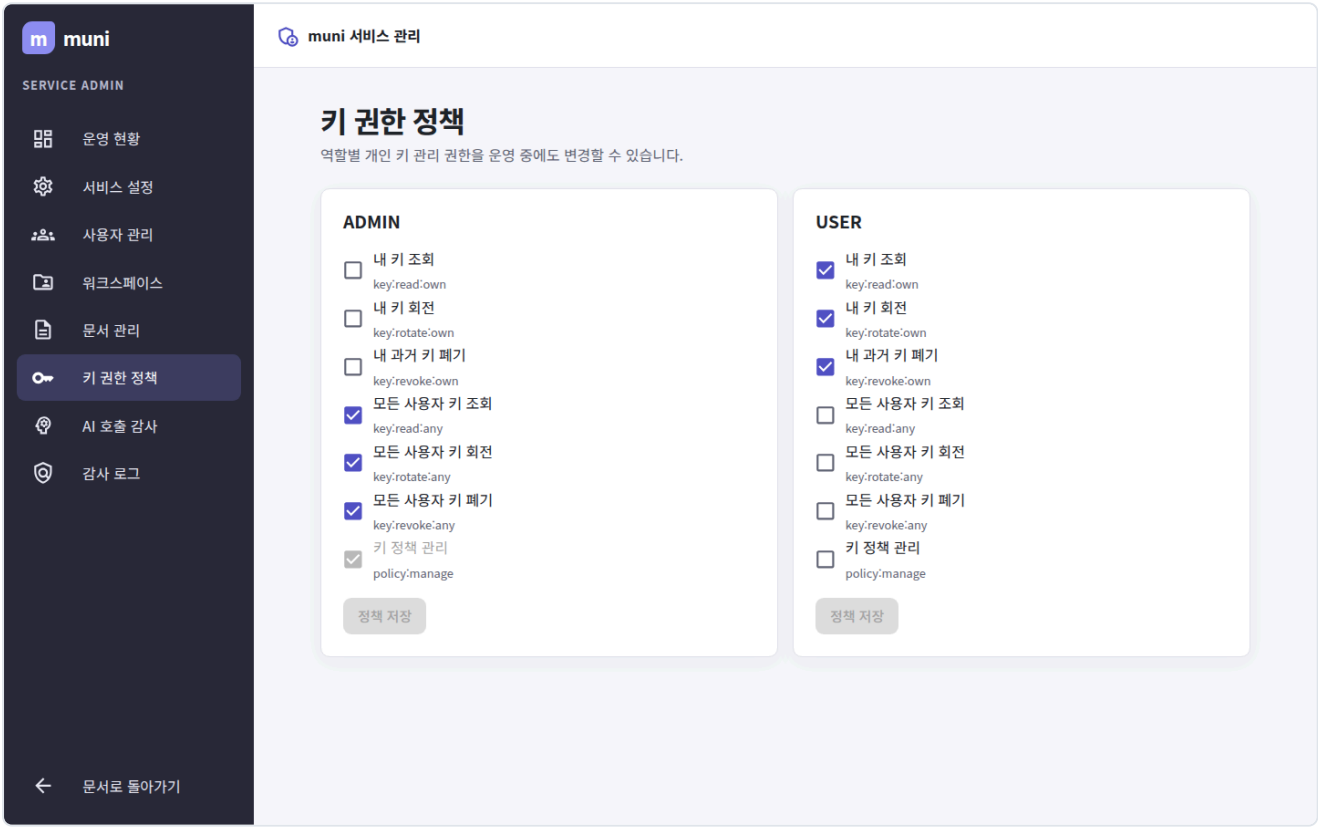
사용자 관리 — 상태·역할·로그인 방식으로 거르고 계정마다 비밀번호·세션·키를 다룬다



계정 만들기 — 한 명씩 만들거나 CSV 로 한꺼번에 가져온다

- 비밀번호를 비워 두면 muni 가 만들어 **그 화면에서 한 번만** 보여 줍니다. 해시로만 보관하므로 나중에 다시 꺼낼 수 없습니다.
- SMTP 를 설정해 두었다면 「접속 정보를 메일로 보내기」로 본인에게 바로 보낼 수 있습니다.
- **남이 정한 비밀번호로는 로그인만 됩니다.** 받은 사람이 바꾸기 전까지 할 수 있는 일은 자기 정보 확인·비밀번호 변경·로그아웃뿐이고, 서버에서 막으므로 API 로 우회할 수 없습니다. 목록의 「임시 비밀번호」 표시가 아직 바꾸지 않은 사람입니다.
- CSV 는 첫 줄에 열 이름을 넣습니다. `email` 만 필수이고 `username` · `displayName` · `role` 은 있으면 씁니다. 한글 열 이름(`이메일` · `이름` · `역할`)도 받습니다. 한 번에 500행, 2MB 까지입니다.
- 계정마다 **비밀번호 재설정 · 로그인 세션 확인과 강제 종료 · 키 관리 · 정리(퇴사자 처리) · 상태 변경**을 할 수 있습니다. 퇴사자 처리는 계정을 지우지 않고 문서와 결재를 넘깁니다 — 절차는 운영 안내의 「퇴사자 정리」에 있습니다.

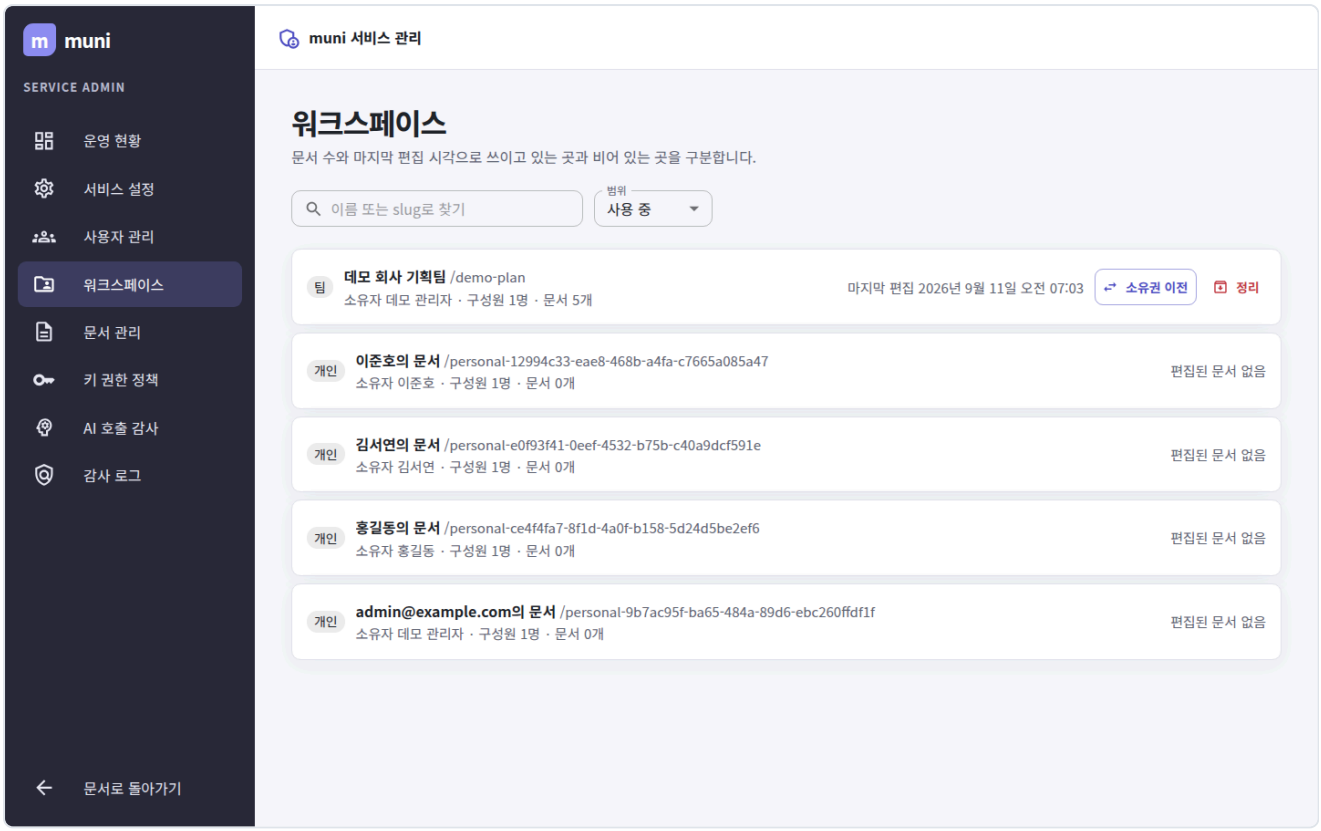
키 권한 정책



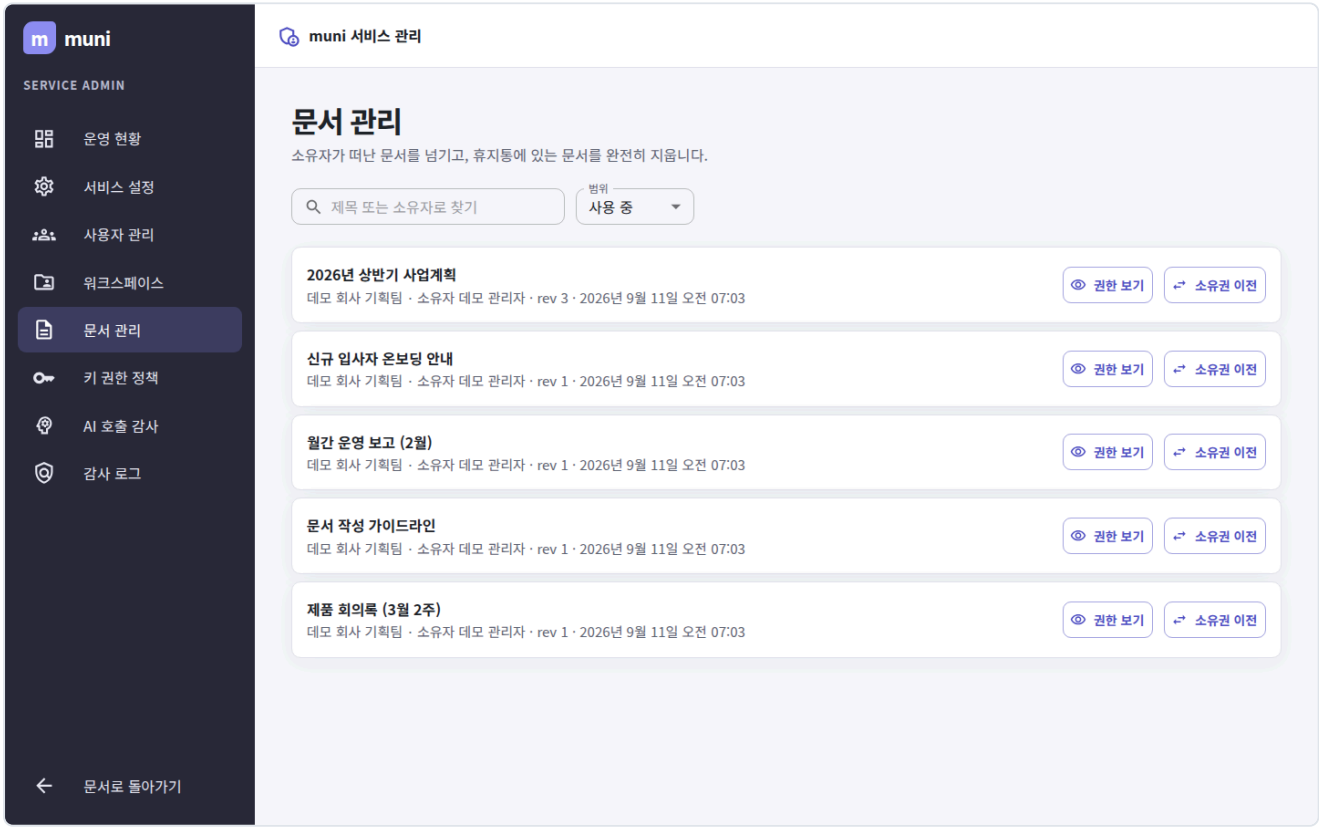
키 권한 정책 — 역할마다 개인 키를 다룰 수 있는 범위를 정한다

사용자별 data key 를 누가 조회·회전·폐기할 수 있는지 역할별로 운영 중에 바꿉니다. 기본값은 `USER` 는 자기 키만 (`key:read:own` · `key:rotate:own` · `key:revoke:own`), `ADMIN` 은 모든 사용자 키(`key:*:any`)와 정책 관리 (`policy:manage`)입니다. `policy:manage` 는 `ADMIN` 에서 뺄 수 없습니다 — 뺄 수 있다면 아무도 정책을 되돌리지 못합니다.

워크스페이스와 문서



워크스페이스 관리 — 워크스페이스마다 소유자와 구성원·문서 수를 보고 소유권을 넘긴다



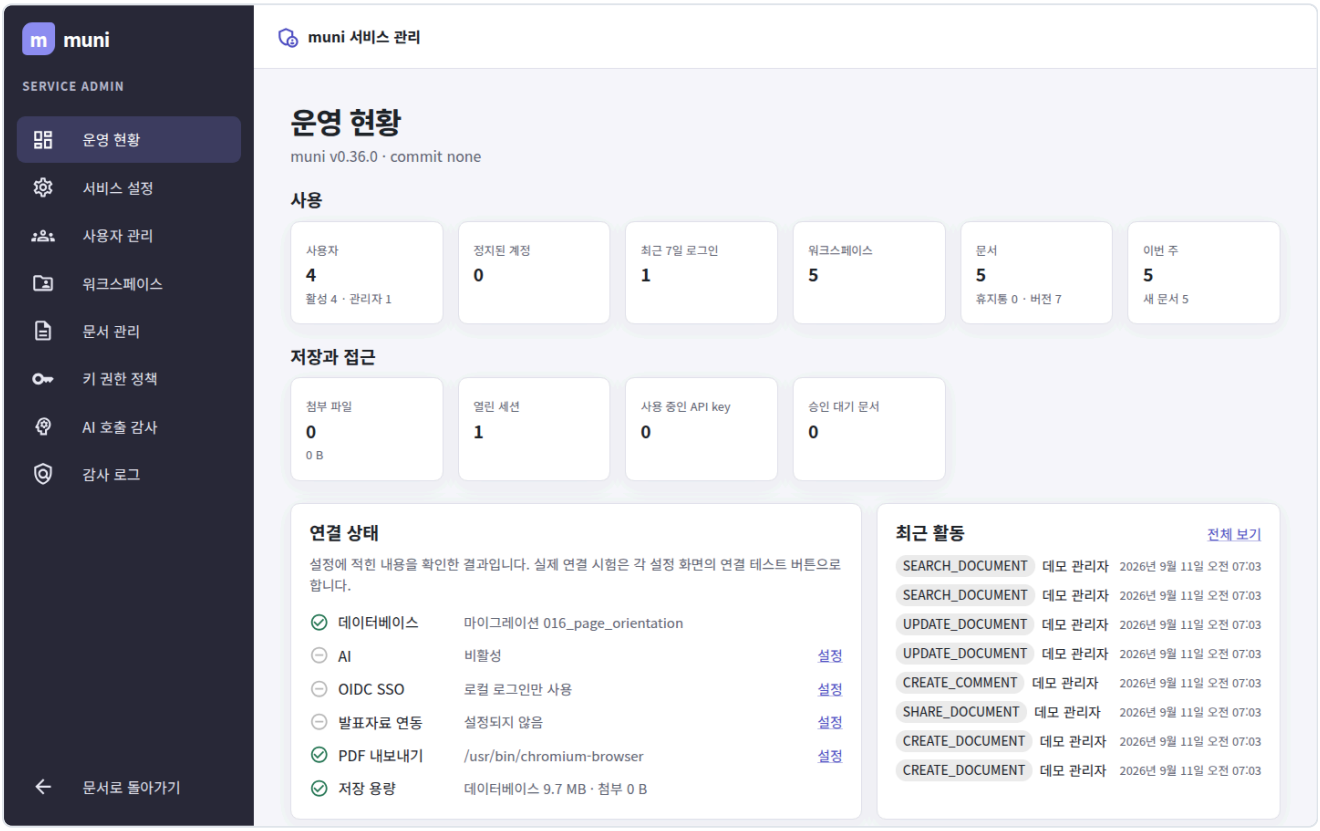
문서 관리 — 소유자가 떠난 문서를 넘기고, 휴지통에 있는 문서를 완전히 지운다

- **워크스페이스** 화면은 문서 수와 마지막 편집 시각으로 쓰이는 곳과 빈 곳을 구분해 줍니다. 「소유권 이전」으로 떠난 사람의 워크스페이스를 넘기고, 「정리」로 쓰지 않는 워크스페이스를 치웁니다(범위를 바꾸면 되살릴 수 있습니다).
- **문서 관리**의 「권한 보기」는 그 문서를 실제로 볼 수 있는 사람 전부를 보여 줍니다 — 직접 공유, 워크스페이스 구성원, 공개 링크까지 한 화면에서 확인합니다. "이 문서 누가 볼 수 있나요" 라는 질문에 답하는 자리입니다.

5. 운영

상태 점검

엔드포인트	메서드	인증	무엇을 답하나
`/healthz`	GET	없음	프로세스가 살아 있는가
`/readyz`	GET	없음	데이터베이스까지 붙어 요청을 받을 수 있는가
`/metrics`	GET	**관리자**	Prometheus 노출 형식
`/api/v1/admin/overview`	GET	관리자	관리 화면 「운영 현황」이 쓰는 요약



운영 현황 — 사용량·저장과 접근·연결 상태·최근 활동을 한 화면에서 본다

「연결 상태」는 설정에 적힌 내용을 확인한 결과이고, 실제 연결 시험은 각 설정 탭의 연결 테스트 단추로 합니다. 「최근 활동」은 감사 로그의 앞부분입니다.

`/metrics` 는 사용자 수·문서 수가 들어 있어 관리자 인증을 요구합니다. Prometheus 로 긁으려면 `api:read` 권한의 관리자 API 키를 만들어 `authorization.credentials` 에 넣습니다. 지표 목록은 운영 안내의 「지표 수집」에 있습니다.

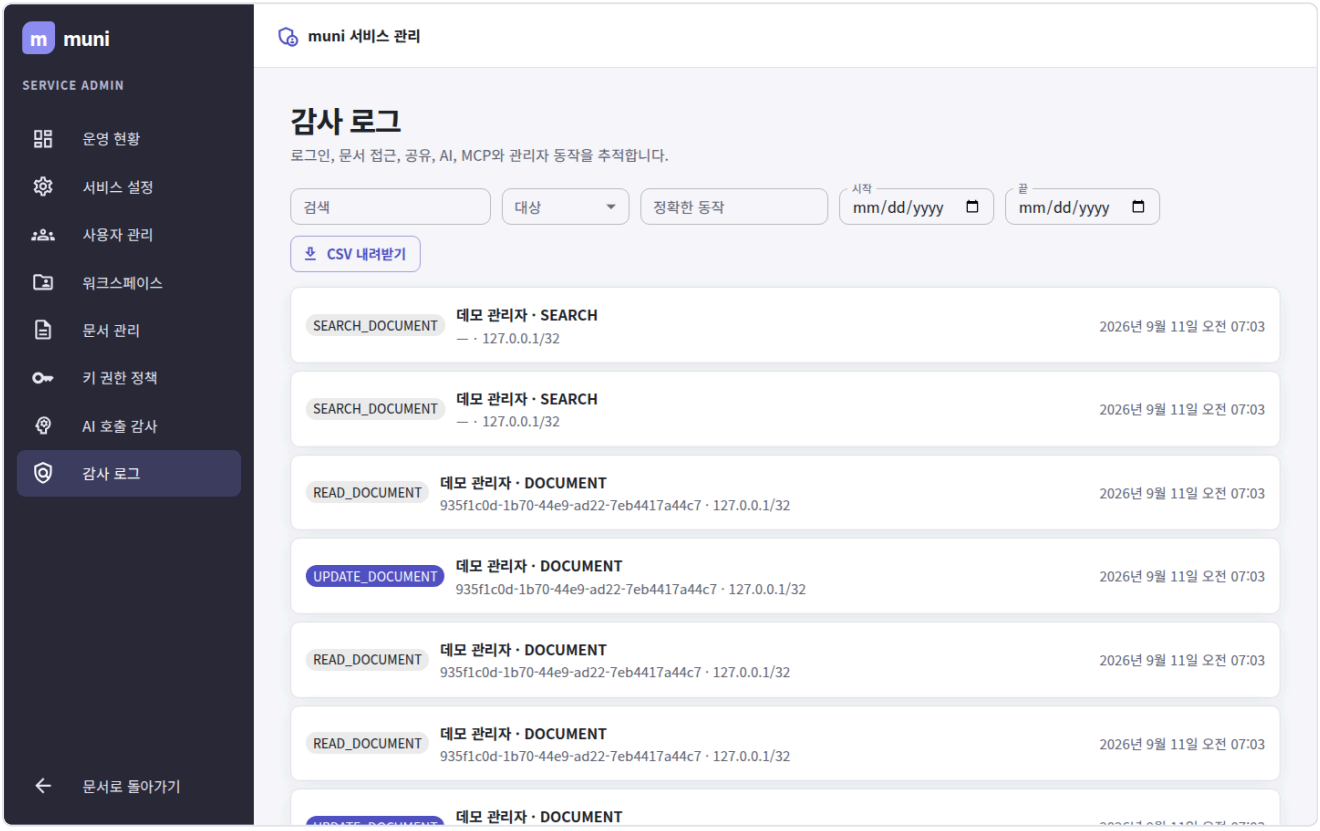
백업

상태는 **데이터베이스와 `ENCRYPTION_KEY` 둘뿐**입니다. 이미지에는 상태가 없습니다.

```
docker compose exec -T postgres \
  pg_dump -U muni -d muni --format=custom --compress=9 \
  > muni-$(date +%Y%m%d).dump
```

키는 백업과 **다른 곳**에 보관합니다. 복원 절차와 "복원해 본 적 없는 백업은 백업이 아니다"는 확인 방법은 운영 안내에 있습니다.

감사 로그



감사 로그 — 대상·동작·기간으로 거르고 CSV 로 내려받는다

로그인, 문서 읽기·수정, 공유, AI·MCP 호출, 관리자 동작이 행위자·대상·IP·시각과 함께 남습니다. 「CSV 내려받기」로 그대로 내보낼 수 있습니다. AI 호출만 따로 보려면 「AI 호출 감사」 화면을 씁니다(AI 를 켜지 않았다면 비어 있습니다). 보존 기간은 서비스 설정의 보존 정책에서 정합니다.

업그레이드

```
gzip -dc muni-v<새 버전>.tar.gz | docker load
# compose.example.yaml 의 image 태그를 새 버전으로 바꾼 뒤
docker compose -f compose.example.yaml --env-file .env up -d
curl -fsS http://127.0.0.1:8080/readyz
```

기동할 때 마이그레이션이 자동으로 적용됩니다. **올리기 전에 백업을 받으세요.** 되돌릴 때는 이미지 태그를 옛 버전으로 되돌립니다 — 다만 마이그레이션은 되돌아가지 않으므로, 새 버전이 스키마를 바꿨다면 **그 백업으로 복원**해야 합니다. 백업보다 새로운 이미지로 복원하는 방향은 되지만, 구버전 이미지로 신버전 데이터를 여는 방향은 지원하지 않습니다.

6. 장애 대응

증상	확인할 곳	조치
컨테이너가 바로 죽는다	`docker compose logs muni` 의 `invalid startup configuration` / `invalid encryption key`	환경 변수 네 개와 키 형식을 확인합니다
기동 중 멈춘다	`database connection failed` / `database migration failed`	DSN, 네트워크, DB 계정의 schema·extension 생성 권한
최초 관리자로 못 들어간다	`bootstrap failed`	`BOOTSTRAP_ADMIN_PASSWORD` 가 12자 이상인지. 계정이 이미 있으면 bootstrap 은 덮어쓰지 않습니다
`/readyz` 만 실패한다	데이터베이스	DB 는 떴는데 muni 가 못 붙는 상태입니다
로그인은 되는데 AI·메일이 안 된다	「운영 현황」의 연결 상태, 각 탭의 연결 테스트	게이트웨이 주소·키. AI 연결 테스트는 실제 호출 endpoint 와 적용된 보정을 함께 보여 줍니다
PDF 내보내기가 느리거나 거절된다	`muni_pdf_renders_in_progress` 와 `muni_pdf_renders_limit`	앞 값이 뒤에 붙어 있으면 줄을 선 것입니다. `MUNI_PDF_CONCURRENCY` 와 메모리를 함께 올립니다
PDF 만 실패한다	로그의 Export 오류, `MUNI_CHROMIUM_PATH`	read-only 컨테이너에서 `/tmp` tmpfs 를 뺐는지 확인합니다
특정 문서만 열리지 않는다	「문서 관리」에서 소유자와 휴지통 여부	소유권 이전 또는 복원
디스크가 찬다	보존 정책, `muni_attachment_bytes`	휴지통·버전 정리는 되돌릴 수 없으니 **미리보기로 규모를 먼저** 봅니다
느려졌다는 말이 나온다	`muni_http_request_duration_seconds` 의 p95	요청 로그(`http request`)에 경로별 `duration_ms` 가 남습니다
결재가 며칠째 안 넘어간다	결재자가 퇴사했는지	운영 안내의 「퇴사자 정리」
누가 무엇을 했는지	감사 로그, CSV 내려받기	시각과 문서 ID 로 좁힙니다

로그는 JSON 한 줄씩 표준출력으로 나갑니다. 모든 요청은 `{"level":"INFO","msg":"http request","method":"GET","path":"/api/v1/...","status":200,"duration_ms":8}` 형태로 남고 (`/healthz` · `/readyz` 는 제외), 기동은 `muni started` , 종료는 `shutdown requested` 입니다.

7. 보안

처음에 반드시 바꿀 것

- `BOOTSTRAP_ADMIN_PASSWORD` — 첫 로그인 뒤 개인 설정에서 바꾸고, `.env` 의 값도 그대로 두지 마세요.
- `ENCRYPTION_KEY` — `.env.example` 의 자리표시자를 그대로 쓰면 안 됩니다. `openssl rand -base64 32`.
- `POSTGRES_DSN` 의 데이터베이스 비밀번호.
- Kubernetes 예제 `deploy/kubernetes/muni.yaml` 의 Secret 은 자리표시자입니다. 실제 배포에서는 외부 secret 관리로 주입하세요.

밖에 열지 말 것

- 컨테이너의 `8080` 을 인터넷에 직접 열지 말고 ingress/reverse proxy 뒤에 두고 TLS 를 거기서 끝냅니다.
- PostgreSQL 포트는 muni 만 닿게 합니다.
- `/metrics` 는 관리자 인증이 걸려 있지만, 그래도 사내 모니터링 망에서만 닿게 두세요.

정책으로 조일 수 있는 것 (서비스 설정 → 보안·내보내기)

- 공개 링크 허용 여부 — 링크를 가진 사람이 누구인지 muni 는 알지 못합니다.
- PDF·DOCX 내보내기 허용 여부.
- 세션 유지 시간, API 키 최대 수명, 업로드 크기 상한, 읽기 감사 로그.
- 로컬 로그인 끄기(일반 탭) — SSO 만 쓰는 조직이면 끕니다.

기본으로 켜져 있는 것

- 비밀번호는 Argon2id 로, 세션·API 토큰은 SHA-256 으로 저장하고 쿠키는 HttpOnly·SameSite 입니다.
- OIDC·AI·SMTP·Ptium 비밀값과 사용자 data key 는 AES-256-GCM 으로 봉인합니다.
- 컨테이너는 비 root·read-only 루트 파일 시스템·capability 전부 drop 으로 돕니다.
- 모든 페이지는 `script-src 'self'` 의 콘텐츠 보안 정책을 답니다. 방문 추적을 켜도 요청마다 다른 nonce 와 스니펫이 적은 출처만 더해지고 `'unsafe-inline'` 은 넣지 않습니다(방문 추적).
- 가져오기는 남이 만든 파일을 읽는 일이라, 압축 폭탄·과도한 XML 중첩을 크기와 깊이로 막고 `javascript:` 같은 링크를 버리며 파일이 가리키는 원격 주소를 대신 내려받지 않습니다(SSRF 방지).

더 깊은 절차: 운영 안내 · 전체 구조: 아키텍처 · API 와 MCP: MCP 안내 와 `/api/openapi.yaml`