

Aria 사용자 매뉴얼

Table of Contents

- 1 🚀 시작하기
 - 1.1 Aria 사용자 매뉴얼
- 2 📊 대시보드
 - 2.1 메인 대시보드
- 3 🔔 알림
 - 3.1 알림 이력
 - 3.2 알림 설정
- 4 ⚙️ 설정·계정
 - 4.1 시스템 설정
 - 4.2 버전 관리
 - 4.3 프로필
 - 4.4 사용자 관리
 - 4.5 IP 접근 관리
- 5 💻 서버 관리
 - 5.1 서버 목록
 - 5.2 그룹 목록
 - 5.3 매니저 목록
 - 5.4 Tomcat 업그레이드
 - 5.5 일괄 배포
 - 5.6 설정 파일 가져오기
 - 5.7 설정 변경 반영

6 ⚡ 에이전트 관리

6.1 시작/정지

6.2 에이전트 대시보드

6.3 보고서

6.4 로그 뷰어

6.5 어플리케이션 배포

6.6 모니터링

6.7 시스템 정보

6.8 CATALINA / JVM

6.9 스레드 목록

6.10 Heap Status

6.11 전역 설정 - Connector

6.12 전역 설정 - Executor

6.13 전역 설정 - Logging

6.14 전역 설정 - DBPool

6.15 전역 설정 - Cluster

6.16 XML 수정 - server.xml

6.17 XML 수정 - context.xml

6.18 XML 수정 - web.xml

6.19 XML 수정 - logging.properties

6.20 설정 변경 이력

6.21 가상 호스트 - Host

6.22 가상 호스트 - Context

6.23 가상 호스트 - Access Log

6.24 가상 호스트 - DBPool

6.25 가상 호스트 - Cluster

1 시작하기

1.1 ARIA 사용자 매뉴얼

제품 버전: ARIA v1.0 최종 수정일: 2026-06-22

ARIA(아리아) v1.0은 Java 기반 WAS 환경에서 소스 수정 없는 에이전트 연동을 통해 서버 제어, 설정 관리, 실시간 모니터링 및 자동 복구 기능을 하나의 웹 콘솔에서 통합 제공하여 운영 효율성과 장애 대응력을 극대화하는 **엔터프라이즈 미들웨어 통합 관리 플랫폼**입니다.

솔인시스템(주)이 자체 개발하여 **GS인증 1등급**을 취득하였으며, 분산된 개별 미들웨어 운영을 웹 GUI 환경으로 일원화하여 운영 편의성과 시스템 안정성을 극대화합니다.

1.1.1 핵심 가치

ARIA는 운영 효율성 향상, 신속한 장애 대응, 안정적인 서비스 유지를 목표로 다음 8가지 핵심 기능을 제공합니다.

-  **WAS 인스턴스 관리**: 수십~수백 대의 미들웨어 인스턴스를 하나의 웹 콘솔 화면에서 일원화하여 관리합니다.
-  **설정 제어 및 배포**: `server.xml`, `context.xml`, `web.xml` 등 WAS 설정 파일을 GUI에서 편집하고 원클릭으로 안전하게 동기화 배포합니다.
-  **실시간 모니터링**: CPU, 메모리, 스레드 풀(Executor), DB 커넥션 풀(DataSource), 응답 속도 등 핵심 성능 지표를 실시간 시각화합니다.
-  **로그 탐색 및 실시간 조회**: 분산된 서버의 로그를 중앙에서 실시간(Tail)으로 스트리밍 조회하고, 원하는 로그 파일 다운로드 및 패턴 분석을 지원합니다.
-  **장애 감지 및 자동 복구 (WatchDog)**: 서버 인스턴스의 상태를 실시간 감지하여 이상 발생 시 자동 재기동 및 관리자 즉시 알람을 수행합니다.
-  **애플리케이션 배포**: 개발된 WAR 파일을 웹 콘솔에서 간편하게 업로드하고 전체 대상 서버에 순차/일괄 배포 및 안전한 롤백을 수행합니다.
-  **스마트 알람 센터**: 임계치를 초과한 리소스 사용이나 이상 징후 발생 시 이메일, 슬랙 등 다양한 채널로 경고 알람을 즉시 전송합니다.
-  **보고서 및 통계**: 수집된 시스템 리소스 및 접속 로그 통계를 기반으로 정기 운영 보고서를 자동으로 생성합니다.

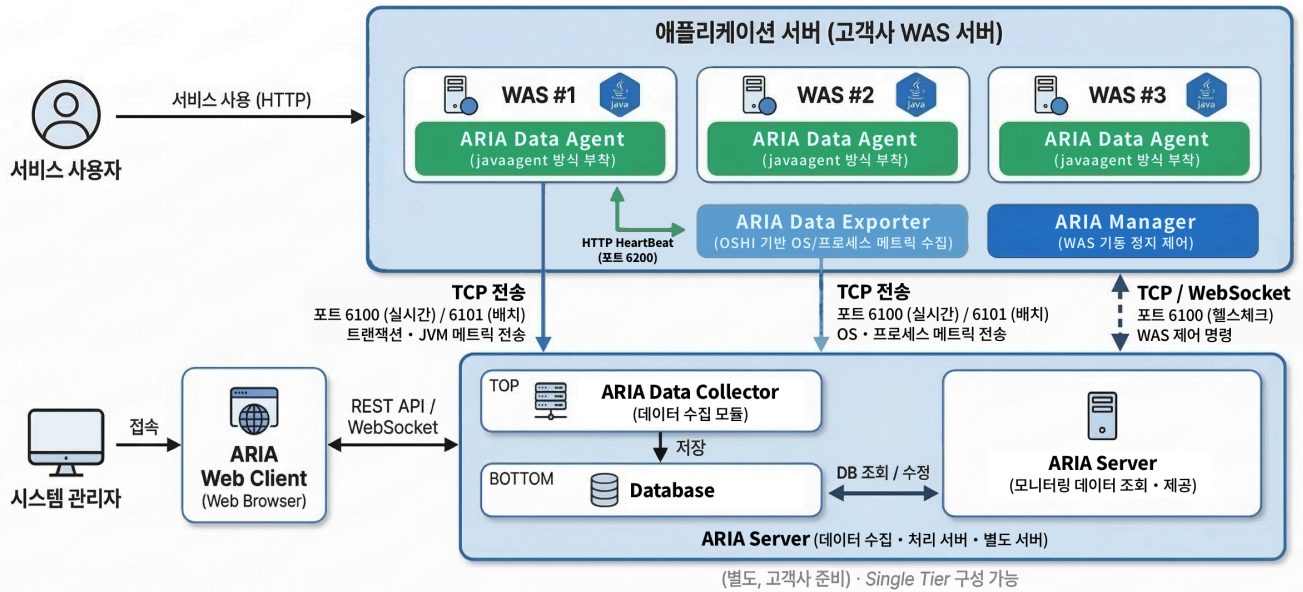
1.1.2 시스템 아키텍처 개요

ARIA는 **중앙 관리 서버(ARIA Server)**와 각 대상 호스트에 설치되는 **ARIA Manager**, **ARIA Data Agent**, **ARIA Data Exporter**로 구성된 분산형 아키텍처입니다.

대상 호스트마다 동일한 구성이 설치되며, 한 호스트에 WAS 인스턴스가 여러 개인 경우 **ARIA Manager 하나**가 `agent.xml`로 모두 관리합니다.

구성 개념도

ARIA v1.0 서비스 구성 개념도



구성 요소	역할	기본 포트	주요 통신
ARIA Server	웹 콘솔, 설정·이력 저장, 수집 데이터 처리·조회	웹 6800, Collector 6100/6101	← Agent·Exporter TCP 수집 / → Manager HTTP·STOMP 제어
ARIA Manager	WAS 시작/정지, 설정 배포, 패치, WatchDog, agent.xml 동기화	6810	← Server STOMP / → 대상 WAS 스크립트·파일 / ↔ Agent WebSocket·HTTP
ARIA Data Agent	JVM -javaagent 계측, 메트릭·트랜잭션·SQL·XLog 수집, 로컬 진단 API	WebSocket 6921	→ Server TCP 수집 / → Exporter Heartbeat / → Manager 인벤토리
ARIA Data Exporter	Agent Heartbeat 등록, OSHI 기반 호스트·프로세스 메트릭 수집 (선택)	6200	← Agent HTTP / → Server TCP 수집

용어 안내: 웹 콘솔에서 **서버(에이전트)**는 ARIA에 등록된 WAS 인스턴스를 뜻합니다. **매니저 목록**의 Manager는 미들웨어 내장 관리 앱이 아니라, 호스트에 설치된 **ARIA Manager** 프로세스입니다.

역할 분리: Agent는 모니터링 데이터를 수집하고, WAS 제어(시작/정지·설정 반영)는 Manager가 담당합니다. Exporter는 Agent와 별도 프로세스로 호스트 OS 메트릭만 수집합니다.

시스템 요구사항

구분	ARIA Server	대상 서버 (Manager · Agent · WAS)
OS	Windows, Linux 공통 지원	Windows, Linux 공통 지원
Java	JDK 1.8 이상	JDK 1.8 이상 (1.8 ~ 17 권장)
지원 미들웨어 (v1.0)	-	Apache Tomcat 8.5, 9.0, 10.1 이상
최소 사양	4 Core CPU / 16GB RAM	대상 WAS 서버 리소스에 포함 (최소 영향)

1.1.3 가이드 바로가기

사용자의 역할과 목적에 따라 가이드를 선택하여 확인하세요.

설치 및 초기 구성

대상 서버에 ARIA Manager·Data Agent를 설치하고, 매니저 등록 후 WAS 인스턴스를 추가·연동하는 방법을 확인합니다.

실무 운영자 가이드

대시보드 모니터링, 인스턴스 시작/정지 제어, 실시간 로그 탐색, 애플리케이션 배포 및 롤백 등 일상적인 운영 방법을 확인합니다.

시스템 관리자 가이드

사용자 계정 생성, 권한 역할 부여, 알림 채널·규칙 설정, 시스템 환경 변수 등 전반적인 플랫폼 정책을 확인합니다.

1.1.4 문의 및 고객 지원

제품 도입 문의 및 기술 지원이 필요한 경우 아래의 창구를 이용해 주시기 바랍니다.

- 공식 웹사이트: www.solinsystem.co.kr
- ARIA 제품 소개: <https://www.solinsystem.co.kr/product/aria.do>
- 대표 전화: 02-6749-6250
- 제품 및 구매 문의: 02-6749-6260 / sales@solinsystem.co.kr

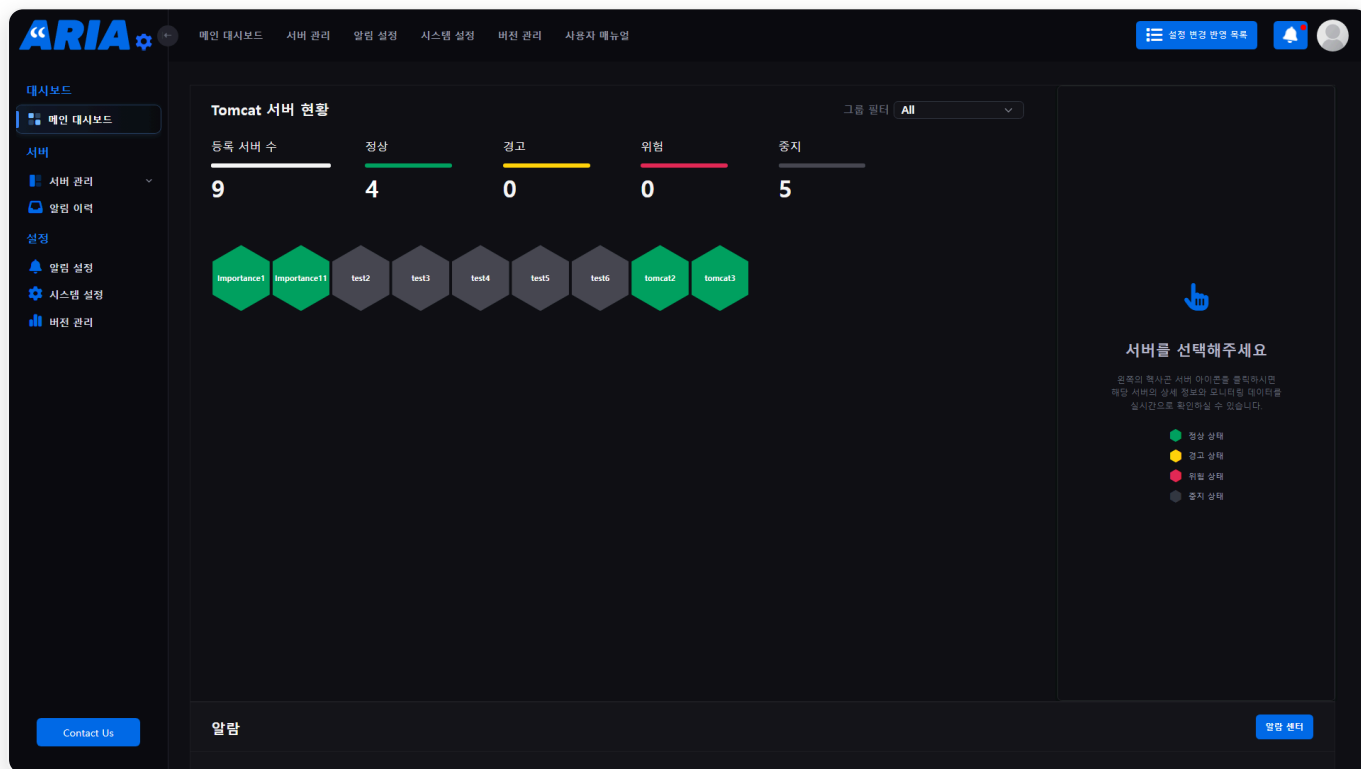
2 대시보드

2.1 메인 대시보드

경로: /

ARIA에 로그인하면 가장 먼저 보이는 메인 화면입니다. 등록된 모든 Tomcat 서버를 **헥사곤(Hexagon)** 형태로 시각화하여 한눈에 전체 인프라의 상태를 파악할 수 있습니다. 헥사곤을 클릭하면 우측에 서버 상세 패널이 열리며, 이 패널에서 모니터링 대시보드 진입, 서버 정보 수정 등 핵심 작업을 바로 수행할 수 있습니다.

2.1.1 화면



2.1.2 1. 헥사곤 맵

화면 중앙에는 등록된 모든 서버가 헥사곤 형태로 배치됩니다. 각 헥사곤의 색상으로 서버의 현재 상태를 즉시 파악할 수 있습니다.

색상	상태	의미
 녹색	Normal	서버가 정상 가동 중

색상	상태	의미
● 노란색	Warning	주의 필요 (리소스 경고 등)
● 빨간색	Critical	즉각 조치 필요
● 회색	Dead/Offline	서버 중지 또는 연결 끊김

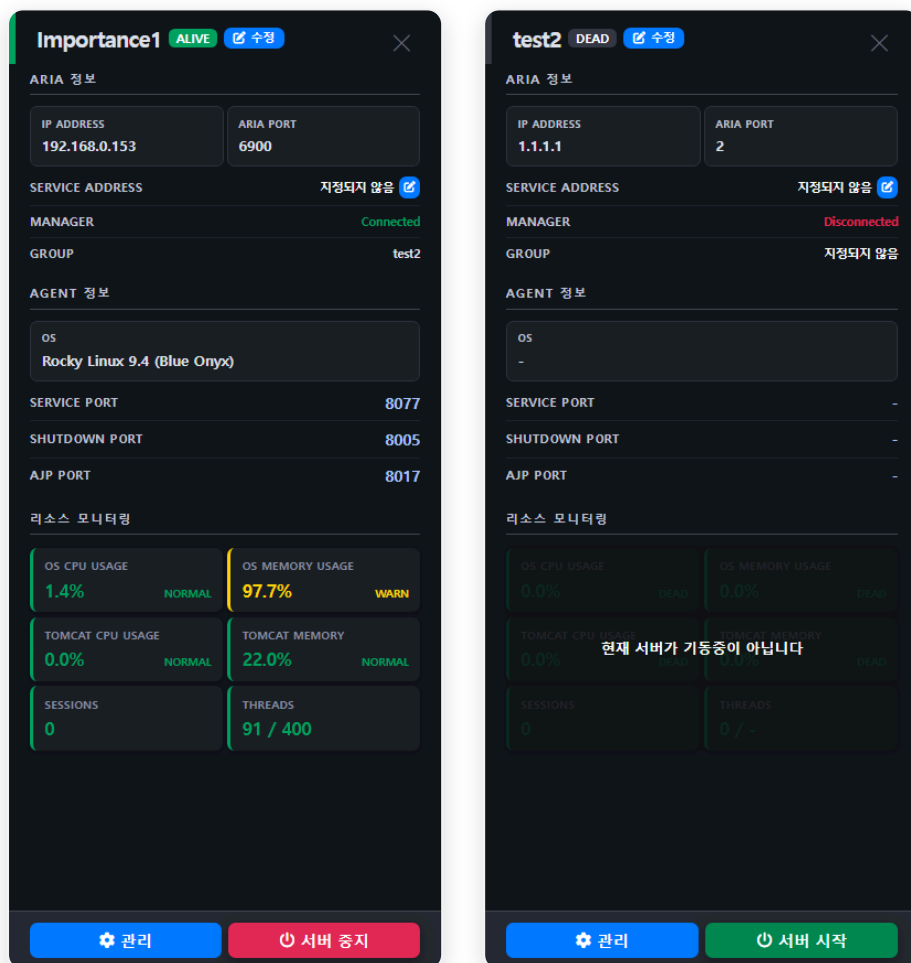
헥사곤 위에는 서버명이 표시되며, 에이전트 연결 상태 및 중요도(Importance) 레벨이 반영됩니다.

알람 센터

우상단의 **알람 센터** 버튼을 클릭하면 발생한 알람 목록 화면으로 이동합니다. 배지 숫자를 통해 미확인 알람 건수를 바로 확인할 수 있습니다.

2. 서버 상세 패널

헥사곤을 클릭하면 화면 우측에 해당 서버의 상세 정보 패널이 슬라이드되어 열립니다.



패널에서는 다음 정보와 기능을 제공합니다.

항목	설명
서버명 / 그룹	등록된 서버 이름과 소속 그룹
상태 배지	현재 연결 및 가동 상태
IP / 포트	서버 접속 주소 정보
에이전트 버전	설치된 ARIA 에이전트 버전
관리창 이동	해당 서버의 에이전트 관리 화면으로 이동
서버 수정	서버 등록 정보(이름, 그룹 등) 수정

서버 정보 수정

패널 내 **수정** 버튼을 클릭하면 서버의 등록 메타 정보를 변경하는 **서버 수정** 모달이 열립니다.

2.1.3 💡 활용 팁

- **빠른 상태 파악:** 매일 업무 시작 시 메인 대시보드를 확인하여 밤 사이 발생한 Dead/Critical 서버를 즉시 파악하세요.
- **hexagon 색상:** 전체가 녹색이면 모든 서버가 정상입니다. 다른 색상의 hexagon은 바로 클릭하여 상세 상태를 확인하세요.
- **알람 센터 연동:** 알람 배지 숫자가 증가하면 알람 센터에서 원인을 확인하고 대응하세요.

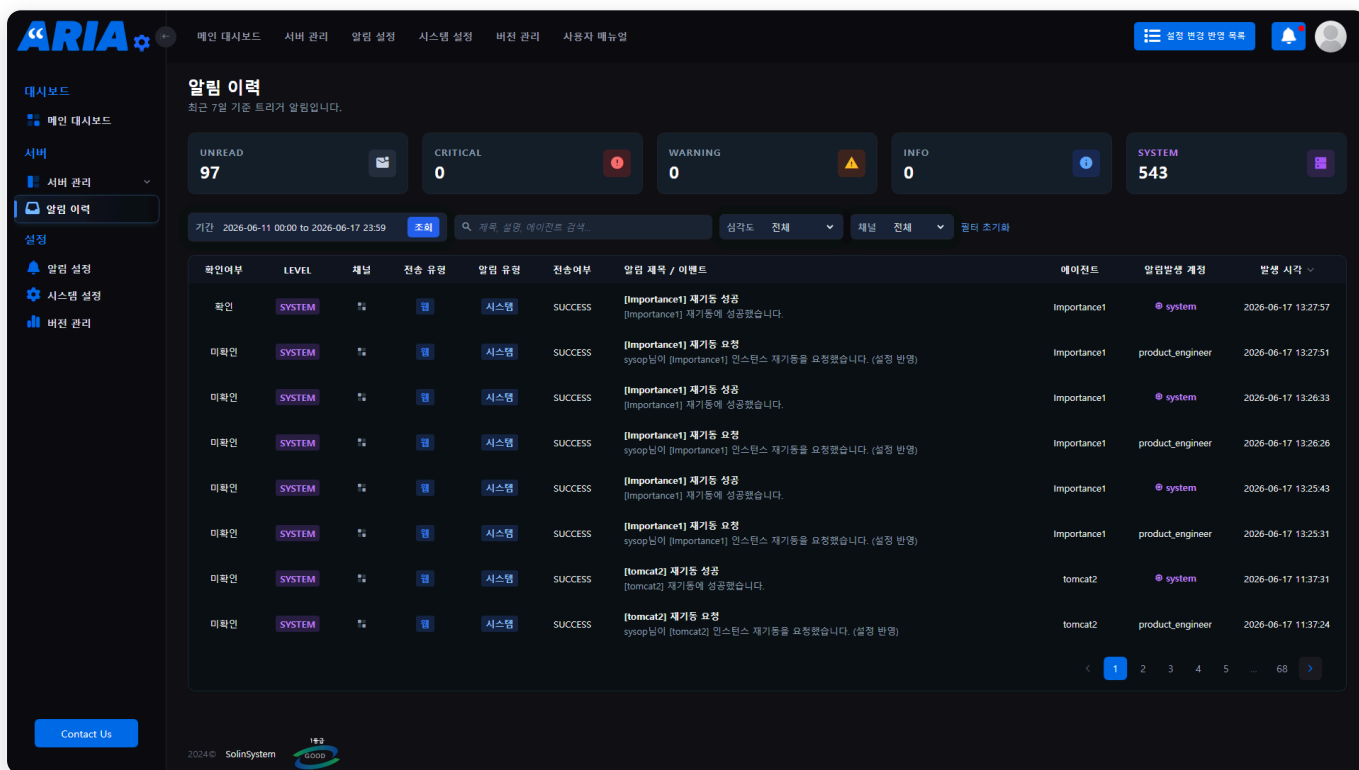
3 알림

3.1 알림 이력

경로: 좌측 사이드바 **알림 관리** > **알림 이력** (구 URL로 접근해도 동일 화면)

알림 센터에서 확인 처리된 알림을 포함하여, 과거에 발생한 모든 알림의 이력을 조회하는 페이지입니다. 특정 기간에 어떤 서버에서 어떤 종류의 이상이 얼마나 발생했는지 분석할 때 활용합니다.

3.1.1 화면



3.1.2 1. 화면 구성

영역	설명
요약 카드	UNREAD, CRITICAL, WARNING, INFO, SYSTEM 건수 표시
조회 조건	기간, 키워드 검색, 심각도, 채널 필터

영역	설명
이력 테이블	조건에 맞는 알림 이력 목록

2. 알림 이력 조회

2-1. 검색 조건

조건	설명
기간	조회 시작일시 ~ 종료일시 설정 후 조회 버튼 클릭 (기본: 최근 7일)
키워드 검색	제목, 설명, 에이전트명으로 테이블 내 즉시 검색
심각도	CRITICAL, WARNING, INFO, SYSTEM 중 선택 (기본: 전체)
채널	Web UI, Slack, Email, SMS 중 선택 (기본: 전체)

필터 초기화 버튼을 누르면 키워드·심각도·채널 필터가 초기화됩니다. 기간을 지정해 조회한 상태에서는 **기본 최근 7일** 조회 화면으로 돌아갑니다.

1. 조회 조건을 설정합니다.
2. 기간을 변경할 때는 **조회** 버튼을 클릭합니다.
3. 조건에 맞는 이력이 테이블에 표시됩니다.

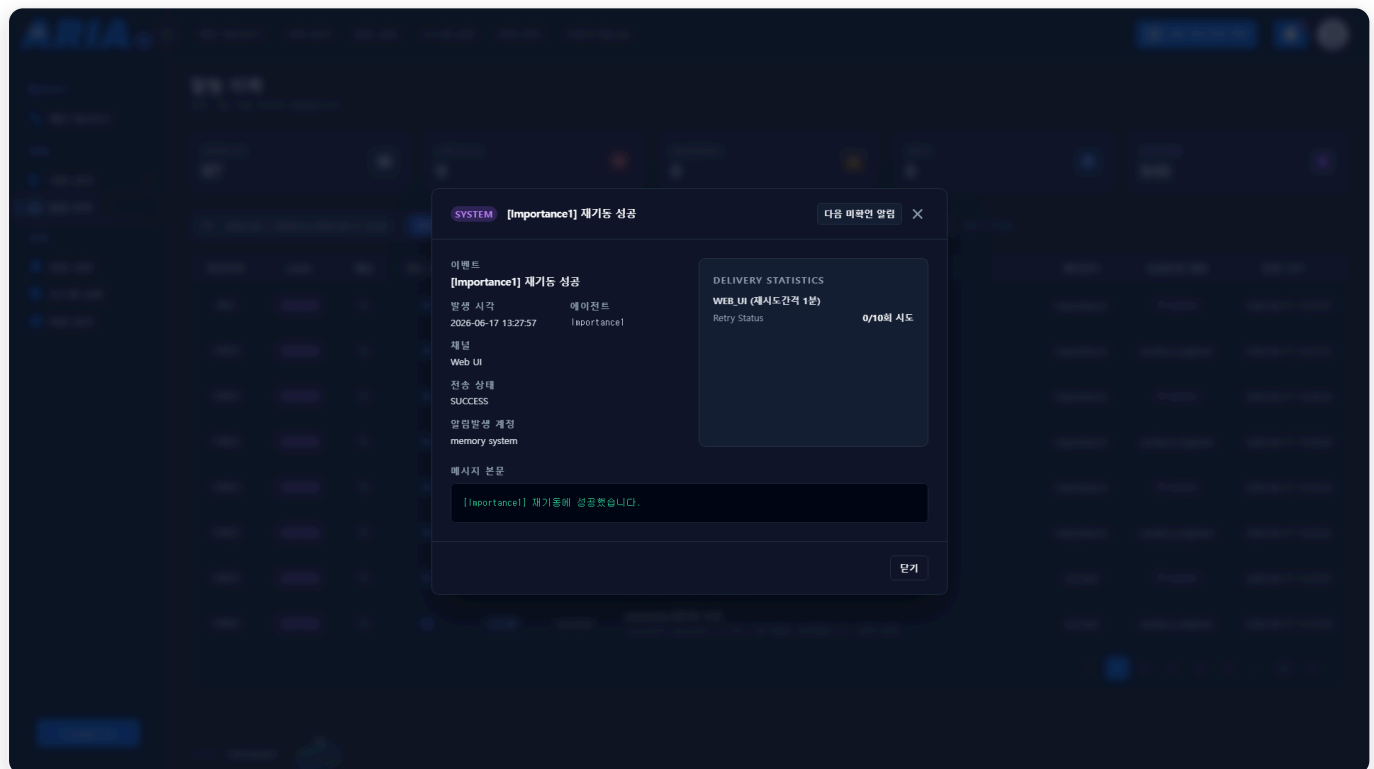
2-2. 이력 테이블 컬럼

컬럼	설명
확인여부	확인 / 미확인
LEVEL	심각도 (CRITICAL, WARNING, INFO, SYSTEM)
채널	알림이 전송된 채널 아이콘
전송 유형	웹 / 채널
알림 유형	시스템 / 트리거
전송여부	전송 상태 (SUCCESS, FAIL, PENDING 등)

컬럼	설명
알림 제목 / 이벤트	알림 제목 및 설명
에이전트	알림이 발생한 대상 서버(에이전트)
알림발생 계정	알림을 트리거한 주체
발생 시각	알림이 기록된 일시

3.1.3 3. 알림 상세 정보 확인

알림 이력 목록에서 특정 행(알림)을 클릭하면 해당 알림의 세부 정보와 발송 통계를 확인할 수 있는 **상세 정보 모달**이 열립니다. 미 확인 알림을 열면 자동으로 읽음(확인) 처리됩니다.



모달에서는 다음 상세 정보들을 조회할 수 있습니다.

3-1. 이벤트 상세 정보 (좌측 영역)

- **이벤트**: 발생한 알림의 명칭 및 이벤트명 (예: `[Importance1] 재기동 성공`)
- **발생 시각**: 알림이 기록된 정확한 날짜 및 시각
- **에이전트**: 알림을 보낸 대상 서버(에이전트) 이름
- **채널**: 알림이 전송된 채널 (예: `Web UI`)
- **전송 상태**: 해당 채널로 알림 전송이 성공했는지 여부 (`SUCCESS` / `FAIL` 등)

- **알림발생 계정:** 알림을 트리거한 주체(누가 발생시켰는지, 예: `memory system` 등)
- **메시지 본문:** 실제 전송된 알림 메시지의 전문

3-2. 전송 통계 (오른쪽 영역 - Delivery Statistics)

알림 채널별 상세 전송 시도 상태 및 재시도(Retry) 정보를 표시합니다.

- **Retry Status:** 채널별로 설정된 재시도 횟수 대비 실제 발송 시도 횟수를 나타냅니다. (예: `0/10회 시도`)
- **재시도 안내:** 수치가 `0` 일 경우, **재시도(Retry) 없이 1회 만에 바로 발송에 성공했음**을 의미합니다.

모달 상단의 **다음 미확인 알림** 버튼으로 아직 확인하지 않은 알림을 순서대로 열어볼 수 있습니다.

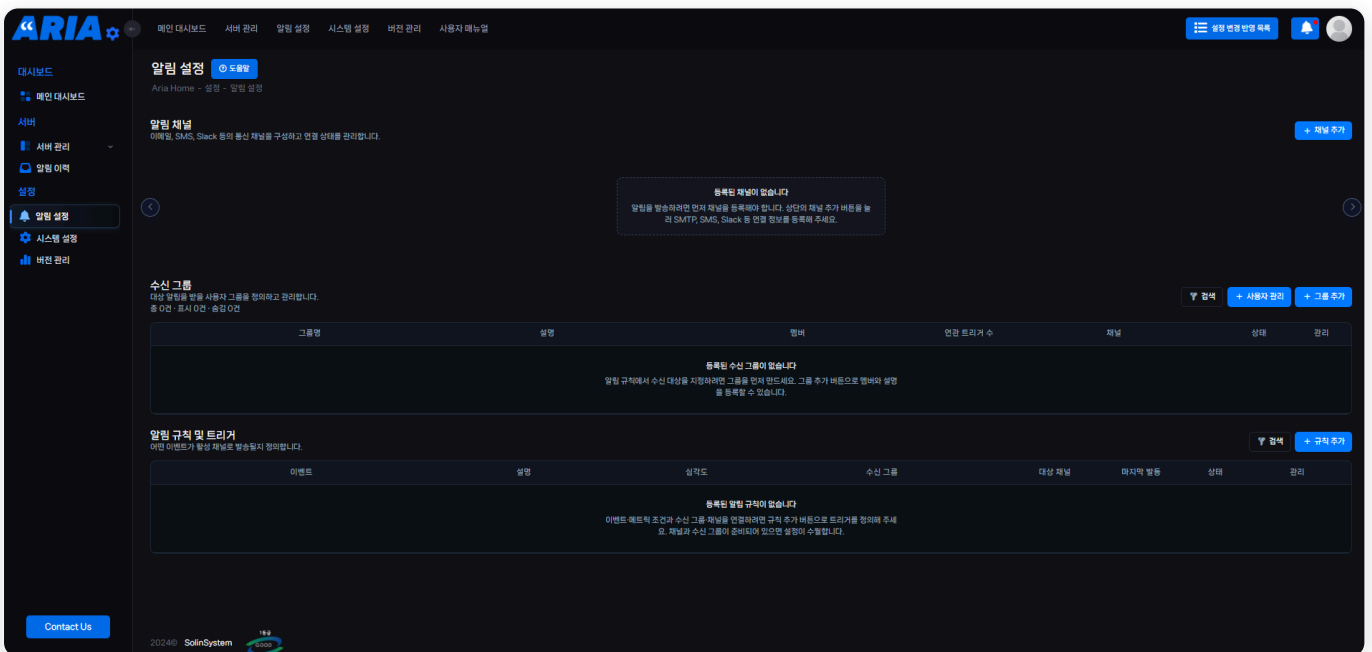
3.1.4 💡 활용 팁

- **장애 패턴 분석:** 월별 알림 이력을 검토하여 특정 서버에서 반복적으로 발생하는 이상 패턴을 파악하고 근본 원인을 분석하세요.
- **미확인 알림 처리:** 상단 UNREAD 카드와 확인여부 컬럼을 참고해 놓친 알림을 빠르게 확인하세요.

3.2 알림 설정

경로: 좌측 사이드바 **알림 > 알림 설정** 메뉴 클릭

ARIA 시스템에서 서버 이상 상태(Dead, Warning 등)가 감지되었을 때, 담당자에게 신속히 상황을 전파할 수 있도록 알림 수신 채널, 수신 사용자/그룹, 그리고 발송 조건에 따른 알림 규칙을 설정하는 종합 관리 화면입니다.



이 화면은 **알림 채널, 사용자 관리, 그룹 관리, 알림 규칙**의 4가지 설정 탭으로 구성되어 있습니다.

3.2.1 1. 알림 채널 설정

서버 이상 발생 시 알림 메시지를 외부 전송 수단으로 내보내기 위한 발송 서버 정보를 연동합니다. ARIA는 이메일(Email), SMS, Slack의 3가지 채널을 지원합니다.

+ 채널 추가

STEP 1 OF 2

Step 1: 채널 설정을 입력하세요.

채널 유형 선택

☒ 이메일

☐ SMS

☐ Slack

좌측에서 등록할 채널 유형을 먼저 선택하세요.

채널 등록 가이드

채널 유형(Email / SMS / Slack)을 선택한 뒤, 우측 폼에서 연결 정보를 입력하고 검증을 완료하세요.

- 유형별 등록 정책: Email 여러 개, SMS 1개, Slack 1개
- 이메일은 용도별(운영/개발/백업 등)로 여러 채널을 추가할 수 있습니다.
- SMS/Slack은 이미 등록되어 있으면 해당 유형 선택이 잠깁니다.

안내: 좌측에서 채널 유형을 선택하면 설정 폼이 표시됩니다.

취소

다음 단계 >

1-1. 이메일(Email) 채널 설정

회사 내부 SMTP 서버 또는 외부 메일 서비스(Gmail 등)를 통해 이메일 알림을 발송하도록 설정합니다.

- **SMTP Host & Port:** 연동할 SMTP 서버의 주소와 포트 번호를 입력합니다. (예: Gmail의 경우 `smtp.gmail.com` / 포트 `587`)
- **보안 프로토콜:** 메일 전송 보안을 위해 **TLS (STARTTLS)**, **SSL**, **없음**의 3가지 방식 중 메일 서버 사양에 맞는 프로토콜을 선택합니다.
- **SMTP 인증 사용:** 메일 발송 서버가 로그인 계정을 요구할 경우 토글을 켜고 사용자 계정(Username)과 비밀번호(앱 비밀번호)를 기입합니다.
- **발신자 정보:** 수신자 메일에 표시될 발신자 이메일 주소와 발신자 이름을 지정합니다.
- **연결 테스트:** 설정을 모두 입력한 후 하단의 **연결 테스트** 버튼을 클릭하여 메일 발송 서버와의 연동 세팅이 올바르게 완료되었는지 즉시 검증할 수 있습니다.

1-2. SMS 채널 설정

서버 긴급 장애 상황을 휴대폰 단문 메시지로 신속히 수신하도록 설정합니다.

채널 추가

Step 1: 채널 설정을 입력하세요.

STEP 1 OF 2

50%

채널 유형 선택

이메일

SMS

Slack

SMS 설정: SMS 제공자 계정을 연결하여 문자 알림을 발송합니다.

SMS 제공자 인증 정보를 입력하세요.

채널명

예: 운영 SMTP

채널 설명

예: 운영 알림 발송용 SMTP 채널

SMS 서비스는 사용량에 따른 과금이 부과될 수 있습니다.

SMS Provider

SOLAPI (솔라피)

* API Key

솔라피 콘솔에서 발급한 API Key

* API Secret

* 발신번호

01012345678 (특수문자 없이)

입력한 인증 정보는 암호화되어 저장됩니다.

연결 테스트

취소

다음 단계

- **SMS Provider:** 현재 SMS 연동은 **SOLAPI(솔라피)** 서비스만 제공하고 있으며, 추후 다양한 제공사(Provider)를 추가 지원할 예정입니다.
- **인증키 입력:** 솔라피 콘솔에서 발급한 **API Key** 및 **API Secret**을 복사하여 입력합니다.
- **발신번호:** 발송처로 등록된 발신번호를 특수문자(-) 없이 숫자만 입력합니다. (예: 01012345678)
- **연결 테스트:** 설정을 마친 후 하단의 **연결 테스트** 버튼을 통해 SMS 전송 기능 작동 여부를 점검할 수 있습니다.
- **참고:** SMS 발송 시 사용량에 따른 과금이 부과될 수 있습니다.

1-3. Slack 채널 설정

협업 툴 Slack의 지정된 채널로 실시간 알림 포매팅 메시지를 수신하도록 설정합니다.

+ 채널 추가

Step 1: 채널 설정을 입력하세요.

STEP 1 OF 2

50%

X

채널 유형 선택

☒

이메일

☐ SMS

☒ Slack

Slack 설정: 워크스페이스와 연동해 채널로 메시지를 전송합니다.

Slack 연결 정보를 입력하세요.

채널명

예: 운영 SMTP

채널 설명

예: 운영 알림 발송용 SMTP 채널

인증 방식

Bot Token (OAuth)

Bot Token

xxxxb-xxxxxxxxxxxxxxxx-xxxxxxxxxxxxxxxx

시스템 기본 공지 채널 (선택)

#general

그룹 수신자에 채널이 없을 때 보조로 쓰일 수 있습니다. 팀 공지는 수신 그룹의 Slack 채널 설정을 권장합니다.

☒ 입력한 인증 정보는 암호화되어 저장됩니다.

☒ 연결 테스트

취소

다음 단계 >

- **인증 방식:** 인증 수단으로 **Bot Token (OAuth)** 방식을 지원합니다.
- **Bot Token:** **Slack API Apps** 개발자 콘솔에서 커스텀 앱을 생성하고 권한(chat:write)을 부여한 후 발급받은 Bot User OAuth Token (xoxb- 로 시작하는 값)을 입력합니다. 상세 절차는 [Slack 공식 문서\(Tokens\)](#)를 참조하세요.
- **시스템 기본 공지 채널:** 알림을 수신할 개별 사용자 또는 그룹 설정에 Slack 채널명이 지정되어 있지 않을 경우, 대체하여 발송될 기본 공지 채널명(예: #general)을 기입할 수 있습니다.
- **연결 테스트:** 토큰 입력 완료 후 **연결 테스트** 버튼을 클릭하여 Slack 봇 앱의 워크스페이스 연결 상태를 검증합니다.

2. 사용자 관리

알림을 수신할 개별 담당자를 등록하고, 해당 사용자의 이메일 주소, 전화번호(SMS 수신용), Slack 멤버 ID 등의 채널별 수신 정보를 관리합니다.

STEP 1 OF 2

50%

+

사용자 관리

Step 1: 사용자 정보를 입력하거나 알림 채널을 설정하세요.

사용자 탐색기

Slack 유저 검색 (팀 공지 채널은 그룹 설정에서 지정)

○ 사용자 검색...

LocalSlack

현재 Local에 등록된 유저정보가 없습니다. 유저를 등록하여 사용해주세요.

1) Local 탭: 우측에서 이름/내부 메모를 입력하고, 필요 채널을 추가한 뒤 저장하세요.

2) Slack 탭: 좌측에서 Slack 유저를 검색/선택하면 해당 정보 기반으로 유저 등록이 준비됩니다.

■ 사용자 정보

이름

이름을 입력해 주세요

내부 메모

예: 프론트엔드 시니어

■ 채널 설정

+ 채널 추가

취소

다음 단계 >

- **Local (DB) 사용자 등록:**

- 우측의 사용자 정보 영역에서 사용자의 **이름**과 **내부 메모(설명)**를 직접 입력합니다.
- 하단의 채널 설정에서 수신받을 채널 정보(Email 주소, Phone 번호, Slack ID)를 활성화된 채널 유형에 맞게 각각 직접 입력하여 등록을 진행합니다.
- 각 채널별 입력란 우측의 **발송 테스트** 기능을 통해 테스트 알림 메시지를 즉시 전송하여 수신자 정보가 올바르게 세팅되었는지 검증할 수 있습니다.

- **Slack 사용자 등록:**

- 좌측의 **사용자 탐색기**에서 **Slack** 탭을 선택한 뒤, 연동된 슬랙 워크스페이스 내의 유저를 검색하고 선택합니다.
- 검색된 슬랙 유저를 선택하면 해당 유저의 정보가 자동으로 연동되어 편리하게 등록할 수 있습니다.

2-2. 사용자 정보 수정

이미 등록된 사용자의 수신 채널 정보나 이름 등을 변경할 수 있습니다.

사용자 관리

Step 2: 최종 확인 후 사용자를 추가하세요.

STEP 2 OF 2

100%

사용자 정보 확인

TE

사용자 식별 정보

test

메모: test

추가된 채널

Email

test@test.com

설정이 확인되었습니다. 사용자를 추가할 준비가 되었습니다.

< 뒤로

확인 및 추가

1. 사용자 관리 모달 좌측의 유저 목록에서 수정하고자 하는 **Local 사용자의 정보(이름)**를 클릭합니다.
2. 우측에 표시되는 정보 및 수신 채널 주소를 수정합니다.
3. 수정 후 **다음 단계**를 거쳐 저장 및 완료하면 사용자 정보가 정상적으로 업데이트됩니다.

3.2.2 3. 그룹 관리

수신 대상으로 등록된 사용자들을 부서, 담당 파트, 혹은 비상 연락망 단위의 **수신 그룹**으로 묶어 체계적으로 관리합니다.

+

그룹 추가

Step 1: 멤버를 선택하고 그룹 정보를 입력하세요.

STEP 1 OF 2

50%

멤버 탐색기

Slack: 채널 목록/멤버 검색 (채널 형 글썽 시
그룹 알림 채널로 지정)

○ 멤버 검색...

Local

Slack

현재 Local에 등록된 유저정보가 없습
니다. 유저를 등록하여 사용해주세요.

1) Local 탭: 우측에서 이름/사부 메오
를 입력하고, 필요 채널을 추가한 뒤
저장하세요.
2) Slack 탭: 좌측에서 Slack 유저를 검
색/선택하면 해당 정보 기반으로 유저
등록이 준비됩니다.

그룹 정보

그룹 이름

예: 프론트엔드 팀

설명 (선택)

그룹의 목적을 간단히 작성해 주세요

Slack 알림 채널 (그룹)

#incidents 또는 C01234ABCDE

좌측 Explorer에서 채널을 클릭하면 여기에 채워집니다. 트리거 시 이 채널로 먼저 전송
됩니다.

☐

그룹 채널 외에 멤버 개인 Slack(DM)으로도 전송

켜면 동일 알림이 채널과 DM에 모두 갈 수 있습니다.

선택된 멤버

0명 선택

선택된 멤버가 없습니다.

취소

다음 단계 >

ARIA

메인 대시보드

서버 관리

알림 설정

시스템 설정

버전 관리

사용자 매뉴얼

알림 설정

도움말

Aria Home

설정

알림 설정

알림 채널

이메일, SMS, Slack 등의 통신 채널을 구성하고 연결 상태를 관리합니다.

채널 추가

수신 그룹

대상 알림을 받을 사용자 그룹을 정의하고 관리합니다.
총 0건 | 표시 0건 | 숨김 0건

검색

사용자 관리

그룹 추가

그룹명	설명	멤버	연관 트리거 수	채널	상태	관리
등록된 수신 그룹이 없습니다 알림 규칙에서 수신 대상을 지정하려면 그룹을 먼저 만드세요. 그룹 추가 버튼으로 멤버와 설명을 등록할 수 있습니다.						

알림 규칙 및 트리거

어떤 이벤트가 활성 채널로 발송될지 정의합니다.

검색

규칙 추가

이벤트	설명	심각도	수신 그룹	대상 채널	관리
등록된 알림 규칙이 없습니다 이벤트 매트릭 조건과 수신 그룹 채널을 연결하려면 규칙 추가 버튼으로 트리거를 정의해 주세요. 채널과 수신 그룹이 준비되어 있으면 설정이 수월합니다.					

규칙 검색

Search rules...

상태

활성 (Active)

비활성 (inactive)

심각도

Critical

Warning

Contact Us

2024 © SolinSystem

한국

0000

3-1. 수신 그룹 구성 (그룹 추가)

사용자 개개인에게 개별 Slack ID를 지정해 메시지를 보낼 수도 있지만, 사용자를 그룹으로 묶어 해당 **Slack 채널(그룹)**로 한 번에 메시지를 전송하게 설정할 수 있습니다.

+

그룹 추가

STEP 1 OF 2

50%

Step 1: 멤버를 선택하고 그룹 정보를 입력하세요.

멤버 탐색기

Slack: 채널 목록·멤버 검색 (채널 탭 클릭 시 그룹 알림 채널로 지정)

○ 멤버 검색...

Local

Slack

테스

테스트



TE

test



● 그룹 정보

그룹 이름

예: 프론트엔드 팀

설명 (선택)

그룹의 목적을 간단히 작성해 주세요

Slack 알림 채널 (그룹)

#incidents 또는 C01234ABCDE

좌측 Explorer에서 채널을 클릭하면 여기에 채워집니다. 트러거 시 이 채널로 먼저 전송합니다.

☐

 그룹 채널 외에 멤버 개인 Slack(DM)으로도 전송
커먼 동일 알림이 채널과 DM에 모두 갈 수 있습니다.

● 선택된 멤버

0명 선택

선택된 멤버가 없습니다.

취소

다음 단계 >

1. + 그룹 추가 버튼을 클릭하여 설정 창을 엽니다.
2. 그룹의 이름, 설명, 그리고 연동할 **Slack 알림 채널(그룹)**명(예: #incidents 또는 채널 ID)을 입력합니다.
3. 좌측의 **멤버 탐색기**(Local 및 Slack 탭 지원)에서 그룹에 소속시킬 유저들을 클릭하여 선택합니다.
4. 선택된 멤버들이 목록에 정상적으로 올라간 것을 확인한 뒤 다음 단계로 이동하여 저장을 완료합니다.

3-2. 수신 그룹 목록 및 관리

등록이 완료된 그룹들은 **수신 그룹** 목록 테이블에서 확인할 수 있습니다.

그룹명	설명	멤버	연관 트리거 수	채널	상태	관리
testgroup		테스트	0		활성	

- **테이블 정보 확인:**
- **멤버:** 그룹에 소속된 멤버 아이콘들이 표시되며, 마우스를 오버하면 멤버의 이름이 툴팁으로 표시됩니다.
- **연관 트리거 수:** 해당 그룹이 알림 규칙에 연결되어 실제로 알림이 발송되도록 매핑된 트리거 규칙의 개수를 나타냅니다.
- **채널:** 이메일, SMS, Slack 등 해당 그룹에 활성화된 채널 아이콘들을 한눈에 확인할 수 있습니다.
- **그룹 수정 및 삭제:**
- **수정:** 관리(Action) 컬럼의 **연필 아이콘**을 클릭하여 그룹의 설정 정보 및 소속 멤버를 수정할 수 있습니다.

- **삭제:** 관리 컬럼의 휴지통 아이콘을 클릭하여 해당 수신 그룹을 완전히 제거할 수 있습니다.

3.2.3 4. 알림 규칙 설정 (트리거 설정)

특정 에이전트(서버)에서 임계치 초과나 특정 동작과 같은 상태 변경 이벤트가 감지되었을 때, 어떤 전송 수단을 통해 어느 그룹에 알림을 발송할지 결정하는 규칙(트리거)을 생성하고 관리합니다.

규칙 생성은 + 규칙 추가 버튼을 클릭하여 진행하며, 총 **4단계**로 구성되어 있습니다.

+

규칙 추가

STEP 1 OF 4

25%

×

Step 1: 규칙의 기본 정보를 입력하세요.

규칙 이름 *

예: CPU 임계값 초과 알림

상세 설명 (선택)

이 규칙에 대한 설명을 입력하세요 (선택 사항)

심각도

!

Critical

즉각 대응 필요

!

Warning

주의 관찰 필요

!

Info

참고용 알림

취소

다음 단계 >

4-1. 1단계: 기본 정보 설정 (심각도 지정)

트리거의 이름, 설명 및 알림 발생 시 표기할 로그 레벨(심각도)을 지정합니다.

+

규칙 추가

STEP 1 OF 4

×

Step 1: 규칙의 기본 정보를 입력하세요.

25%

규칙 이름 *

test

상세 설명 (선택)

이 규칙에 대한 설명을 입력하세요 (선택 사항)

심각도

!

Critical

즉각 대응 필요

!

Warning

주의 관찰 필요

!

Info

참고용 알림

취소

다음 단계 >

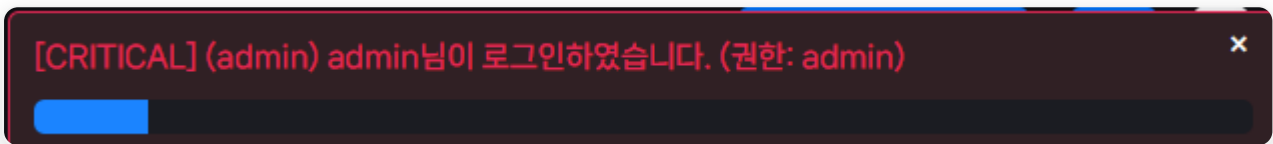
- **규칙 이름 & 상세 설명:** 규칙을 식별할 수 있는 직관적인 이름과 세부 동작 설명을 기입합니다.
- **심각도 (Severity):** 알림의 긴급 수준을 지정합니다. 선택한 심각도에 따라 알림 이력 목록에 출력되는 로그 배지 색상이 다르게 표시됩니다.
- **SYSTEM** (시스템 코어 알림): 중요한 기본 정보에 대하여 사용자의 별도 설정 없이도 상시 출력되는 가장 기본적인 로그 레벨로, 시스템 내에서 가장 높은 우선순위를 가집니다. 주로 다음과 같은 핵심 서버 라이프사이클 및 시스템 상태 감시 시 강제 발생합니다.

분류	설명
서버 라이프사이클	인스턴스 기동 확인 (헬스체크)
	인스턴스 중지 확인
	기동 요청 (사용자가 직접 수행한 경우 포함)
	중지 요청
	강제 종료
	재기동 성공/실패
	포트 중복 기동 실패

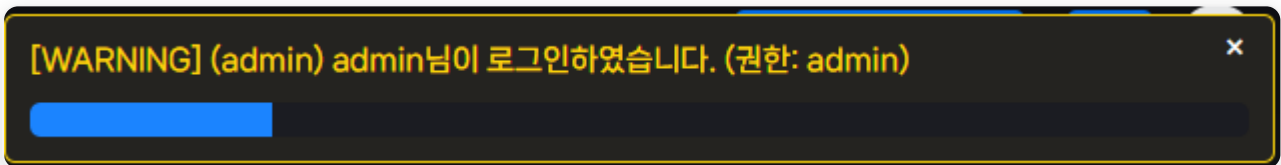
26

분류	설명
Watchdog	복구 성공
	복구 실패
	Watchdog 미실행
그룹	수신 그룹 추가·수정·삭제

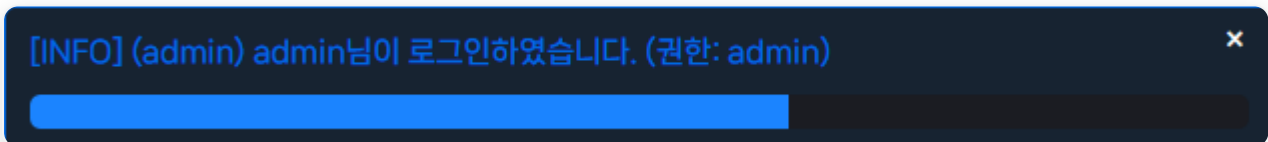
- **Critical** (즉각 대응 필요): 빨간색 배지로 출력



- **Warning** (주의 관찰 필요): 노란색 배지로 출력



- **Info** (참고용 알림): 파란색 배지로 출력



4-2. 2단계: 알림 발송 조건 구성 (Metric 및 Event 조건)

규칙이 발동될 정밀한 감시 조건을 설계합니다.

+ 규칙 추가

STEP 2 OF 4

Step 2: 조건을 선택하고 발동 로직을 구성하세요.

카테고리 탐색기

로직 소스를 선택하세요

Metric
Instance 포함

Audit/Function
이벤트 기반

조건 구성

규칙 발동 로직을 구성합니다.

AND 조건 OR 조건

01 Audit Event

TARGETACTIONACTOR

All중지 확인 (HEALTH)전체 대상

[전체 대상]에서 [중지 확인 (HEALTH)]을(를) 발생시켰을 때 알림 발생

AND

02 Audit Event

TARGETACTIONACTOR

All전체 설정 수정전체 대상

[전체 대상]에서 전체 대상이(가) [전체 설정 수정]을(를) 발생시켰을 때 알림 발생

+ 조건 추가

임시 저장 목록

test오전 10:24

< 뒤로

임시 저장

다음 단계 >

규칙 추가

STEP 2 OF 4

50%

Step 2: 조건을 선택하고 발동 로직을 구성하세요.

카테고리 탐색기

로직 소스를 선택하세요

Metric

Instance 포함

Audit/Function

이벤트 기반

조건 구성

규칙 발동 로직을 구성합니다.

AND 조건

OR 조건

01 Metric Condition

TARGET	PROPERTY	OPERATOR	VALUE
All	OS CPU 사용률 (%)	>	예: 80

[전체 대상]에서 [OS CPU 사용률 (%)]이(가) > 0 일 때 알림 발생

OR

02 Additional Condition

TARGET	PROPERTY	OPERATOR	VALUE
All	OS CPU 사용률 (%)	>	예: 80

[전체 대상]에서 [OS CPU 사용률 (%)]이(가) > 0 일 때 알림 발생

조건 추가

임시저장 목록

저장된 임시 데이터가 없습니다.

< 뒤로

임시 저장

다음 단계 >

- **조건 조합 Logic:** 복수의 조건을 추가할 때 **AND 조건** 또는 **OR 조건**의 토글 스위치 조합을 선택하여 복잡한 트리거 로직을 구현할 수 있습니다.
- **임시 저장 기능:** 작성 중인 세팅 값을 날리지 않고 모달 하단의 **임시 저장** 버튼을 클릭해 보관할 수 있습니다. 보관된 값은 좌측 하단의 **임시/저장 목록**에 기록되어 필요 시 즉시 다시 불러올 수 있습니다.

[!WARNING] 임시 저장 정보는 **서버 메모리 캐시 기반**으로 작동하므로, ARIA 서버가 재기동되면 임시저장 목록의 데이터는 모두 소멸됩니다.

💡 지원 조건 상세 목록

① METRIC(성능 지표) 기반 조건 목록

서버 하드웨어 및 WAS(Tomcat/JVM)의 실시간 수집 수치 임계치를 기준으로 알림을 생성합니다. (>, >=, <, <=, ==, != 등의 비교 연산자 지원)

그룹	조건 항목	지원 비교 연산자	비고
OS 리소스	OS CPU 사용률 (%)	>, <, >=, <=, ==, !=	
	OS 메모리 사용률 (%)	>, <, >=, <=, ==, !=	
	OS 스왑 사용률 (%)	>, <, >=, <=, ==, !=	

그룹	조건 항목	지원 비교 연산자	비고
	디스크 사용률 (%)	>, <, >=, <=, ==, !=	
	네트워크 수신 (Rxbyte)	>, <, >=, <=, ==, !=	
	네트워크 송신 (Txbyte)	>, <, >=, <=, ==, !=	
JVM 리소스	JVM CPU 사용률 (%)	>, <, >=, <=, ==, !=	Tomcat 프로세스 지표
	JVM 메모리 사용률 (%)	>, >=, <, <=, ==, !=	Tomcat 프로세스 지표
	Heap 사용량	>, <, >=, <=, ==, !=	
	Heap 최대량	>, <, >=, <=, ==, !=	
	PermGen/Metaspace 사용량	>, <, >=, <=, ==, !=	
	가동 시간 (h)	>, <, >=, <=, ==, !=	소수점 시간 환산 (예: 1.5h)
	가동 시간 (m)	>, <, >=, <=, ==, !=	누적 분 (예: 360m)
Thread	전체 스레드 수	>, <, >=, <=, ==, !=	
	실행 중인 스레드 (Runnable)	>, <, >=, <=, ==, !=	
	차단된 스레드 (Blocked)	>, <, >=, <=, ==, !=	
	대기 중인 스레드 (Waiting)	>, <, >=, <=, ==, !=	
	시간 제한 대기 (Timed Waiting)	>, <, >=, <=, ==, !=	
	교착 상태 스레드 (Deadlocked)	>, <, >=, <=, ==, !=	데드락 감지 개수
	신규 스레드 (New)	>, <, >=, <=, ==, !=	
	종료된 스레드 (Terminated)	>, <, >=, <=, ==, !=	
Request / Session	요청 수 (Count)	>, <, >=, <=, ==, !=	
	에러 수 (Error)	>, <, >=, <=, ==, !=	
	평균 응답 시간 (ms)	>, <, >=, <=, ==, !=	

그룹	조건 항목	지원 비교 연산자	비고
	세션 수	>, <, >=, <=, ==, !=	
DataSource	Active 커넥션	>, <, >=, <=, ==, !=	DB 커넥션 풀 활성 수
	Idle 커넥션	>, <, >=, <=, ==, !=	DB 커넥션 풀 유휴 수

② AUDIT EVENT(이벤트) 기반 조건 목록

서버 라이프사이클이나 설정 변경, 사용자 관리 등 ARIA 감사 로그에 기록된 특정 시스템 액션을 감지하여 즉시 발동합니다. (발생 시 연산자로 설정)

그룹	감지 이벤트	비고
서버 상태	중지 확인 (HEALTH)	중지 요청·강제 종료도 함께 감지
	기동 확인 (HEALTH)	기동 요청·Tomcat 기동 감지도 함께 포함
인스턴스 관리	인스턴스 추가	
	인스턴스 삭제	
	인스턴스 정보 수정	
사용자 / 시스템	사용자 추가	
	사용자 삭제	
	사용자 수정	
	비밀번호 변경	
	시스템 설정 변경	
	사용자 로그인	
그룹 설정	그룹 추가	
	그룹 수정	
	그룹 삭제	

그룹	감지 이벤트	비고
매니저 / 보안	매니저 추가	
	매니저 수정	
	매니저 삭제	
	IP 차단 추가	
	IP 차단 삭제	
Watchdog	Watchdog 복구 시도	
	Watchdog 성공	
	Watchdog 실패	복구 실패·미실행·오류·잠금 등 포함
	Watchdog 활성화됨	
	Watchdog 비활성화됨	
서버 제어	기동 요청	사용자가 직접 기동 요청 시
	중지 요청	사용자가 직접 중지 요청 시
	기동 실패(포트 중복) (HEALTH)	
배포	애플리케이션 배포 성공	
	애플리케이션 배포 실패	
설정 파일	전체 설정 수정	설정 적용·복원·JVM 환경 반영 등 포함
	Connector 수정	추가·삭제·적용·복원 포함
	Executor 수정	추가·삭제·적용·복원 포함
	Logging 수정	추가·삭제·적용·복원 포함
	DBPool 수정	추가·삭제·적용·복원 포함
	Cluster 수정	추가·삭제·적용·복원 포함

그룹	감지 이벤트	비고
VirtualHost / Context	VirtualHost Host 수정	추가·삭제·적용 포함
	VirtualHost Context 수정	추가·삭제·적용 포함
	VirtualHost Access Log 수정	추가·삭제·적용 포함
	VirtualHost DBPool 수정	추가·삭제·적용 포함
	VirtualHost Cluster 수정	추가·삭제·적용 포함

4-3. 3단계: 수신 그룹 및 채널 지정

조건이 발동했을 때 알림을 전달할 수신처와 전송 재시도 정책을 구성합니다.

- **수신 그룹 다중 선택:** 알림을 수신할 수신 그룹들을 좌측 목록에서 복수로 지정할 수 있습니다.
- **그룹별 채널 커스터마이징:** 특정 그룹마다 이메일, SMS, Slack 등 발송을 원하는 수단을 활성화할 수 있습니다. (아직 채널 연동 정보가 설정되어 있지 않은 수단은 선택할 수 없습니다.)

- **Web UI 단독 발송:** 외부 채널(이메일·SMS·Slack)을 지정하지 않고도 **전역 채널 · Web UI 사용** 설정을 커서 ARIA 웹 콘솔 내의 실시간 토스트/알림 배치로만 이력을 쌓도록 구성할 수 있습니다.
- **실패 시 재시도 정책:** 외부 채널 전송 실패(네트워크 장애 등)에 대비하여 **실패 시 자동 재시도** 토글을 활성화하여 상세 재시도 옵션을 설정합니다.
- **재시도 횟수:** 3회, 5회, 10회 중 선택 가능합니다. (0회는 재시도 없음 — 1회만 시도)
- **재시도 간격:** 1분, 5분, 15분 중 선택 가능합니다. 지정된 시간 간격이 경과할 때마다 재발송을 시도합니다.
- **예시:** 3회 / 5분으로 설정한 경우, 최초 전송 실패 후 5분 간격으로 최대 3회까지 재시도를 수행하며, 최종 실패 시 전송 실패 상태로 이력에 기록됩니다.

4-4. 4단계: 최종 검토 및 메시지 템플릿 확인

설정 정보를 최종 검수하고 실제 채널에서 전송될 메시지 서식을 미리 확인합니다.

- 등록된 최종 조건 조합과 수신 그룹 요약을 확인합니다.
- 오른쪽 미리보기 영역에서 각 채널 유형별로 가공되어 발송될 템플릿(메시지 본문 형태)을 사전에 최종 검토한 뒤 완료합니다.

3.2.4 5. 알림 규칙(트리거) 목록 및 관리

수신 그룹
대상 알림을 받을 사용자 그룹을 정의하고 관리합니다.
총 1건 · 표시 1건 · 숨김 0건

검색
+ 사용자 관리
+ 그룹 추가

그룹명	설명	멤버	연관 트리거 수	채널	상태	관리
testgroup			1		활성	

알림 규칙 및 트리거
이번 이벤트가 활성 채널로 발송될지 정의합니다.

검색
+ 규칙 추가

이벤트	설명	심각도	수신 그룹	대상 채널	마지막 발생	상태	관리
info-test	사용자 로그인 발생 시	Info			2026-06-19 09:59:30	활성	
test	중지 확인 (HEALTH) 발생 시 외 1개	Warning	testgroup		미발동	활성	
warn-test	사용자 로그인 발생 시	Warning			2026-06-19 09:58:44	비활성	
critical-test	사용자 로그인 발생 시	Critical			2026-06-19 09:59:00	비활성	

- **트리거-그룹 상호 연계 확인:**
- `main.png` 메인 테이블 화면에서 각 트리거(규칙)에 어떤 수신 그룹들이 포함되어 있는지 목록 컬럼을 통해 한눈에 파악할 수 있습니다.
- 역으로, **그룹 관리** 목록의 **연관 트리거 수** 컬럼을 통해 특정 그룹이 몇 개의 트리거 규칙에 등록되어 실제 알림을 대기 중인지 교차 검증(Cross-Check)할 수 있습니다.
- **ON/OFF 토글을 통한 빠른 전환:**
- 테이블 가장 앞쪽(좌측)에 있는 **ON/OFF 토글 스위치**를 클릭하여, 별도의 편집 페이지 진입 없이도 트리거 규칙을 즉시 활성화 하거나 비활성화할 수 있습니다.
- **트리거 수정 및 제거:**
- **수정:** 관리(Action) 컬럼의 **연필 아이콘**을 클릭하여 마법사를 통해 조건, 수신 그룹, 재시도 횟수 등의 설정을 자유롭게 편집할 수 있습니다.
- **제거:** 관리(Action) 컬럼의 **휴지통 아이콘**을 클릭하여 불필요해진 알림 규칙을 영구적으로 삭제할 수 있습니다.

3.2.5 💡 활용 팁

- **연결 테스트 필수 수행:** 채널을 추가할 때는 오타 및 인증키 에러 방지를 위해 반드시 저장 전에 하단의 **연결 테스트**를 수행해 정상 연동 여부를 점검하세요.
- **다중 채널 구성:** 중대한 장애(Dead) 규칙에는 Slack뿐만 아니라 SMS 채널을 함께 묶어 긴급 전파율을 높이길 권장합니다.

4 설정·계정

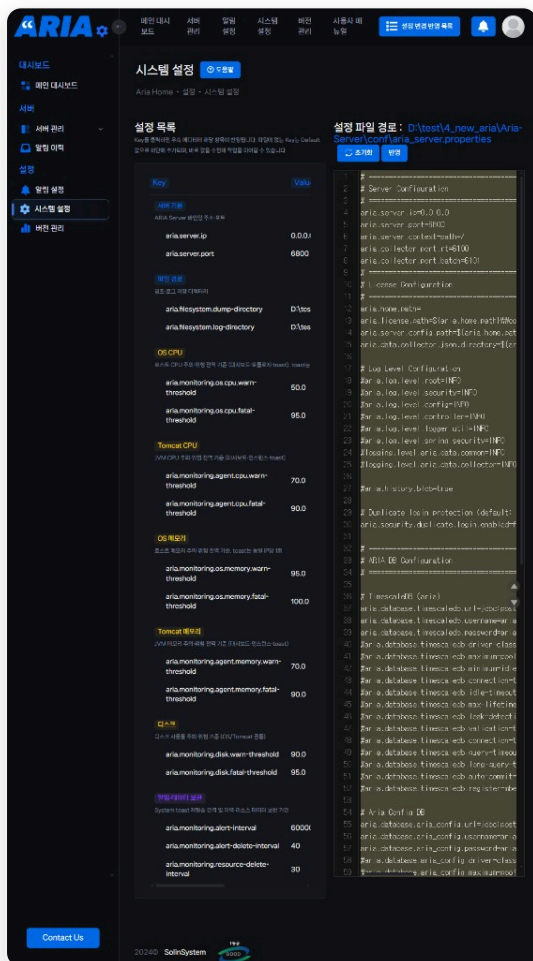
4.1 시스템 설정

경로: 좌측 사이드바 **설정·계정** > **시스템 설정**

시스템 설정은 ARIA Server 전체에 적용되는 전역 환경 파일인 `aria_server.properties` 를 관리하는 화면입니다. 좌측 **설정 목록**에서 주요 Key를 빠르게 찾고, 우측 **에디터**에서 전체 파일을 직접 편집할 수 있습니다.

관리자 전용: 시스템 설정 변경은 관리자(admin) 이상 권한이 필요합니다. 잘못된 설정 변경은 ARIA 서비스 전체에 영향을 줄 수 있으므로 주의하여 변경하세요.

4.1.1 화면 구성



화면은 좌우 2열로 구성됩니다.

영역	설명
설정 목록 (좌측)	자주 쓰는 Key를 그룹별로 정리한 표. Key, Value, Default 열로 현재 값과 기본값을 비교할 수 있습니다.
설정 파일 경로 (우측 상단)	실제 파일 경로(예: <code>conf/aria_server.properties</code>)와 초기화, 반영 버튼
에디터 (우측)	<code>aria_server.properties</code> 전체 내용을 편집하는 코드 에디터

목록 상단 안내:

Key를 클릭하면 우측 에디터에 해당 항목이 반영됩니다. 파일에 없는 Key는 Default 값으로 하단에 추가되며, 바로 값을 수정해 작업을 이어갈 수 있습니다.

4.1.2 1. 설정 목록 사용법

1-1. Key 클릭

- **파일에 이미 있는 Key:** 에디터에서 해당 줄로 이동하고 파란색 하이라이트로 강조됩니다.
- **파일에 없는 Key:** 주석(있는 경우)과 함께 Default 값으로 **파일 하단에 추가**됩니다. 기존 스크롤 위치를 유지한 채 하단에 붙이므로, 에디터를 맨 아래에서 작업 중이어도 위로 튀지 않습니다.

1-2. 톨팁

Key 위에 마우스를 올리면 한국어 설명 톨팁이 표시됩니다. 각 항목의 용도와 대시보드 알림에 미치는 영향을 확인할 수 있습니다.

1-3. 그룹 구분

설정 목록은 아래 그룹으로 나뉩니다.

그룹	포함 Key (요약)
서버 기본	<code>aria.server.ip</code> , <code>aria.server.port</code>
파일 경로	<code>aria.filesystem.dump-directory</code> , <code>aria.filesystem.log-directory</code>
OS CPU	<code>aria.monitoring.os.cpu.warn-threshold</code> , <code>aria.monitoring.os.cpu.fatal-threshold</code>
Tomcat CPU	<code>aria.monitoring.agent.cpu.warn-threshold</code> , <code>aria.monitoring.agent.cpu.fatal-threshold</code>
OS 메모리	<code>aria.monitoring.os.memory.warn-threshold</code> , <code>aria.monitoring.os.memory.fatal-threshold</code>

그룹	포함 Key (요약)
Tomcat 메모리	aria.monitoring.agent.memory.warn-threshold, aria.monitoring.agent.memory.fatal-threshold
디스크	aria.monitoring.disk.warn-threshold, aria.monitoring.disk.fatal-threshold
알림	aria.monitoring.alert-interval

에디터에는 위 목록에 없는 다른 설정(SMTP, DB, 라이선스 등)도 그대로 포함되어 있으며, 에디터에서 직접 수정할 수 있습니다.

4.1.3 2. 모니터링 임계값 (aria.monitoring.*)

aria.monitoring.* 임계값은 ARIA 전역에서 주의(Warning)·위험(Critical/Fatal) 상태를 판정하는 가장 기초적인 기준입니다.

2-1. 적용 범위

구분	적용 화면·기능
OS CPU·메모리	메인 대시보드 hexagon, 토폴로지, 상세 패널, System metric 웹 toast
Tomcat CPU·메모리	메인 대시보드, 인스턴스 대시보드, 상세 패널, System metric 웹 toast
디스크	모니터링 화면 등 OS/Tomcat 공통 주의·위험 판정

반영 후 서버 메모리 캐시(MonitoringThresholdHolder)가 즉시 갱신됩니다. 메인 대시보드(GET_MAIN_DATA)·인스턴스 대시보드(GET_AGENT_RESOURCE) 폴링 응답에 monitoringThresholds 스냅샷이 포함되므로, 열린 대시보드 탭은 새로고침 없이 다음 폴링(약 3~10초)에 임계값·색상이 동기화됩니다.

2-2. 기본값 (Default)

Key	Default	단위
aria.monitoring.os.cpu.warn-threshold	50.0	%
aria.monitoring.os.cpu.fatal-threshold	90.0	%
aria.monitoring.agent.cpu.warn-threshold	70.0	%
aria.monitoring.agent.cpu.fatal-threshold	90.0	%

Key	Default	단위
<code>aria.monitoring.os.memory.warn-threshold</code>	90.0	%
<code>aria.monitoring.os.memory.fatal-threshold</code>	100.0	%
<code>aria.monitoring.agent.memory.warn-threshold</code>	70.0	%
<code>aria.monitoring.agent.memory.fatal-threshold</code>	90.0	%
<code>aria.monitoring.disk.warn-threshold</code>	90.0	%
<code>aria.monitoring.disk.fatal-threshold</code>	95.0	%

2-3. System metric 웹 toast와의 관계

동일한 임계값이 **브라우저 웹 UI toast**(System metric 알림)에도 사용됩니다.

구분	동작
OS metric (CPU·메모리)	동일 IP(호스트)당 1회만 발행. toast에 해당 IP의 서버 이름 이 함께 표시됩니다.
Tomcat metric (JVM CPU·메모리)	인스턴스(에이전트)별로 발행됩니다.
<code>aria.monitoring.alert-interval</code>	동일 System metric toast 재발송 최소 간격 (밀리초, 기본 600000 = 10분). 대시보드 색상·주의/위험 판정과는 별도 설정입니다.

알림 설정과의 구분: 알림 설정의 알림 규칙은 사용자가 정의한 조건(Dead, Warning 등)에 따라 이메일·SMS·Slack 등 **외부 채널**로 발송합니다. 시스템 설정의 **모니터링 임계값**은 **화면 색상·상태 판정·웹 toast**용 전역 기준이며, 두 설정은 역할이 다릅니다.

4.1.4 3. 설정 초기화

초기화 버튼을 클릭하면 현재 편집 중인 에디터 내용을 **저장하지 않고** 마지막으로 서버에서 불러온 내용으로 되돌립니다. 실수로 잘못 입력한 경우 **반영** 전에 초기화로 원복할 수 있습니다.

주의: 초기화는 화면에서 편집 중인 내용만 취소합니다. 이미 **반영**된 설정은 되돌리지 않습니다.

4.1.5 4. 설정 반영

반영 버튼을 클릭하면 에디터의 내용이 `aria_server.properties` 파일에 저장되고 ARIA Server에 즉시 적용됩니다.

- 저장 후 모니터링 임계값은 서버 메모리 캐시에 **즉시 reload**됩니다.
- 반영이 완료되면 에디터 내용이 서버에서 다시 로드됩니다.

시스템 메일 SMTP (`aria.mail.*`)

`aria.mail.*` Key는 **임시 비밀번호 발송** 등 ARIA 시스템이 직접 보내는 메일에 사용됩니다. **알림 설정의 이메일 채널 SMTP**와는 **별개**이며, 서버 이상 알림 발송 설정과 공유되지 않습니다.

Key	설명
<code>aria.mail.host</code> , <code>aria.mail.port</code>	SMTP 서버 주소·포트
<code>aria.mail.username</code> , <code>aria.mail.password</code>	SMTP 인증 계정·비밀번호
<code>aria.mail.sender</code>	발신자 이메일
<code>aria.mail.target</code>	임시 비밀번호 수신 이메일 (미입력 시 username 사용)
<code>aria.mail.auth</code> , <code>aria.mail.enabletls</code>	SMTP 인증·TLS 사용 여부

변경 후에는 사용자 관리 등에서 **임시 비밀번호 메일 발송**이 정상 동작하는지 확인하세요. 알림 설정의 연결 테스트로는 이 설정을 검증할 수 없습니다.

4.1.6 💡 활용 팁

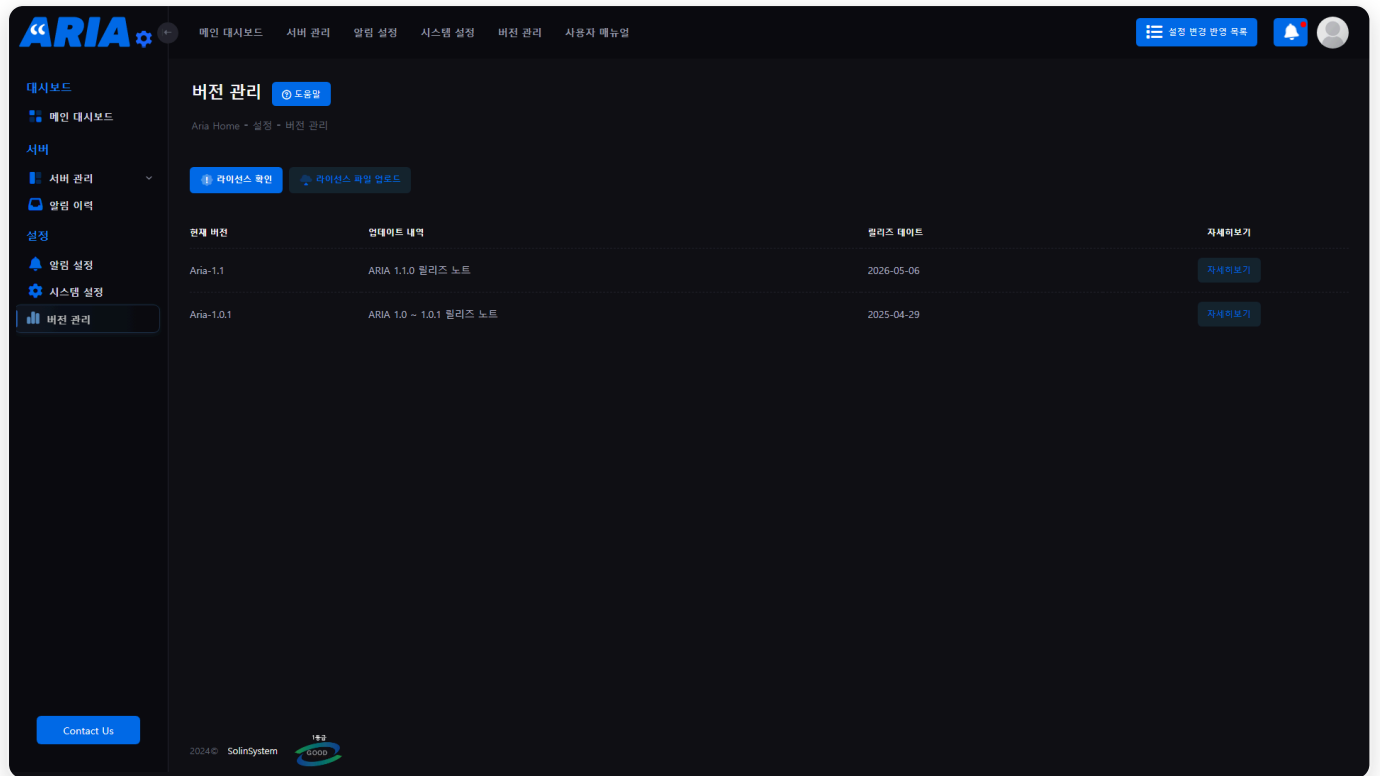
- **변경 전 백업**: 중요 설정 변경 전 현재 값을 별도로 기록해 두면 문제 발생 시 복구에 도움이 됩니다.
- **목록 + 에디터 병행**: 자주 쓰는 Key는 목록에서 클릭해 위치를 잡고, 세부 값·주석은 에디터에서 직접 수정하세요.
- **임계값 조정**: 네트워크·부하 특성에 맞게 warn/fatal을 조정한 뒤 대시보드를 새로고침해 hexagon·게이지 색상 변화를 확인하세요.
- **시스템 메일 vs 알림 채널**: `aria.mail.*` 는 임시 비밀번호 등 **시스템 메일**용입니다. 서버 이상 알림용 SMTP는 **알림 설정**에서 별도로 구성합니다.

4.2 버전 관리

경로: 좌측 사이드바 **설정·계정 > 버전 관리**

현재 설치된 **ARIA 버전 정보와 라이선스 유효기간을 확인**하고, 라이선스 파일을 업로드하여 갱신하는 화면입니다. ARIA는 라이선스 기반 제품이므로, 라이선스가 만료되면 일부 기능이 제한될 수 있습니다.

4.2.1 화면



4.2.2 1. ARIA Server 버전 정보 확인

자세히 보기 버튼(ARIA Server 항목)을 클릭하면 현재 설치된 ARIA Server의 상세 버전 정보를 확인할 수 있습니다.

- ARIA Server 버전
- 빌드 날짜
- Java 버전

4.2.3 2. ARIA Agent 버전 정보 확인

자세히 보기 버튼(ARIA Agent 항목)을 클릭하면 연결된 각 에이전트의 버전 정보를 확인할 수 있습니다.

- 에이전트 버전이 ARIA Server 버전과 달라도 동작하지만, 가능하면 동일 버전으로 유지하는 것을 권장합니다.

4.2.4 3. 라이선스 현황 확인

라이선스 확인 버튼을 클릭하면 현재 라이선스의 상세 정보를 확인합니다.

항목	설명
라이선스 적용 회사명	라이선스에 등록된 고객사

항목	설명
라이선스 만료일	기능 제한이 시작되는 일자
허용 에이전트 수	최대 등록 가능한 서버 수

라이선스 만료일이 가까워지면 솔인시스템에 갱신 라이선스를 요청하세요.

4.2.5 4. 라이선스 파일 업로드

새 라이선스 파일을 발급받은 경우, **라이선스 파일 업로드** 버튼을 클릭하여 파일을 선택하고 업로드합니다. 업로드가 완료되면 새 라이선스가 즉시 적용됩니다.

라이선스 갱신 절차: 1. 솔인시스템(sales@solinsystem.co.kr)에 라이선스 갱신 요청 2. 발급받은 라이선스 파일(`.lic` 또는 `.license`)을 저장 3. 이 화면에서 **라이선스 파일 업로드**로 적용

4.2.6 💡 활용 팁

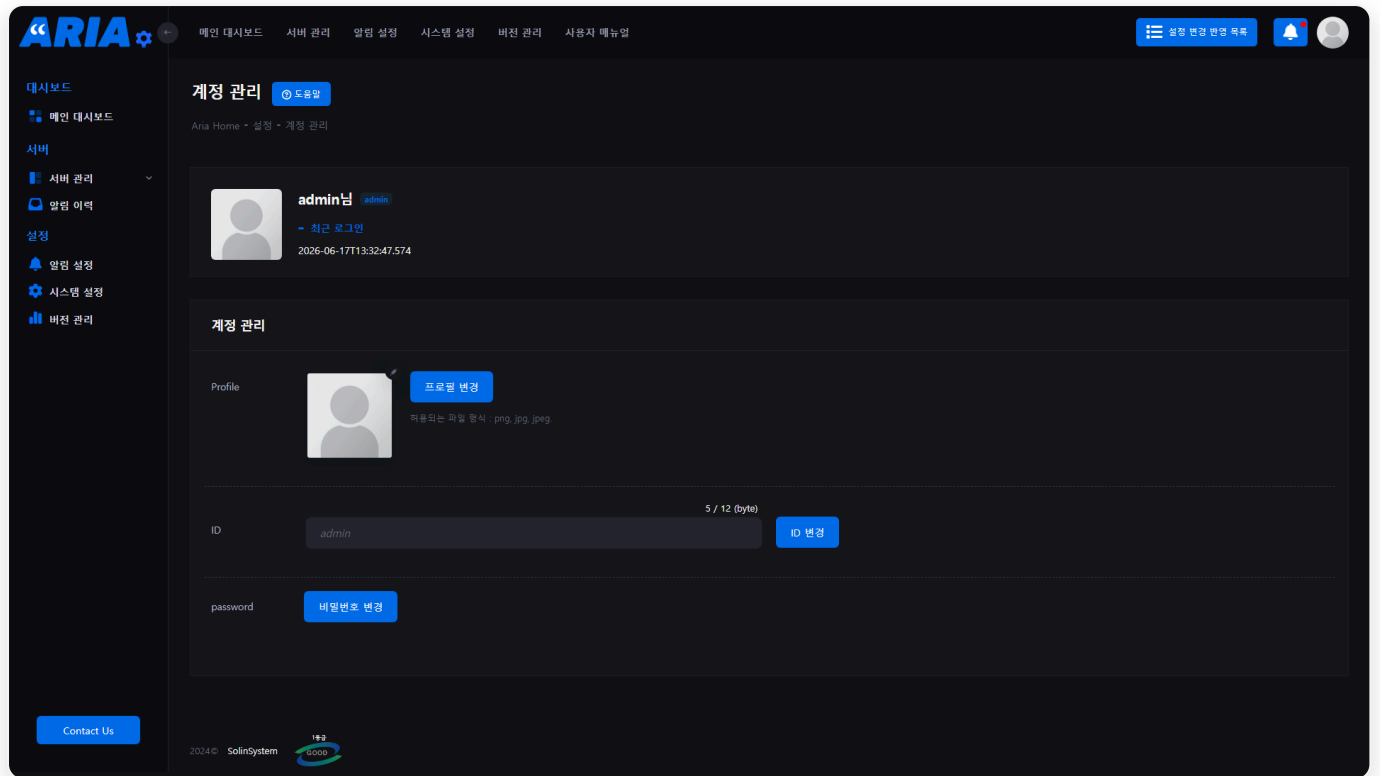
- **에이전트 버전 점검:** **서버 목록**에서 Dead 서버가 많을 경우 에이전트 버전 불일치 여부를 이 화면에서 먼저 확인하세요.
- **만료 알림:** 만료 30일 전부터 갱신 절차를 시작하여 서비스 중단을 방지하세요.

4.3 프로필

경로: 헤더 우측 **프로필 메뉴** > **내 프로필** (또는 좌측 사이드바 **설정·계정** > **프로필**)

현재 로그인한 **내 계정 정보를 확인하고 수정**하는 화면입니다. 프로필 이미지 변경, 로그인 ID 변경, 비밀번호 변경을 이 화면에서 모두 처리할 수 있습니다.

4.3.1 화면



4.3.2 1. 계정 관리 탭

계정 관리 탭에서 현재 계정의 이름, 이메일, 권한 등 기본 정보를 확인할 수 있습니다.

2. 프로필 이미지 변경

프로필 변경 버튼을 클릭하면 프로필 이미지를 업로드하거나 변경할 수 있습니다.

- 지원 파일 형식: JPG, PNG (권장 크기: 200×200px 이상)

4.3.3 3. 로그인 ID 변경

ID 변경 버튼을 클릭하면 현재 사용 중인 로그인 아이디를 변경할 수 있습니다.

- 변경 후에는 새 아이디로 다시 로그인해야 합니다.
- ID 변경 시 현재 비밀번호 입력이 필요합니다.

4.3.4 4. 비밀번호 변경

비밀번호 변경 버튼을 클릭하면 현재 비밀번호 확인 후 새 비밀번호를 설정할 수 있습니다.

항목	설명
현재 비밀번호	본인 확인용
새 비밀번호	변경할 비밀번호
새 비밀번호 확인	오타 방지를 위한 재입력

보안 권고: 비밀번호는 영문 대소문자·숫자·특수문자를 조합하여 8자리 이상으로 설정하세요.

4.3.5 💡 활용 팁

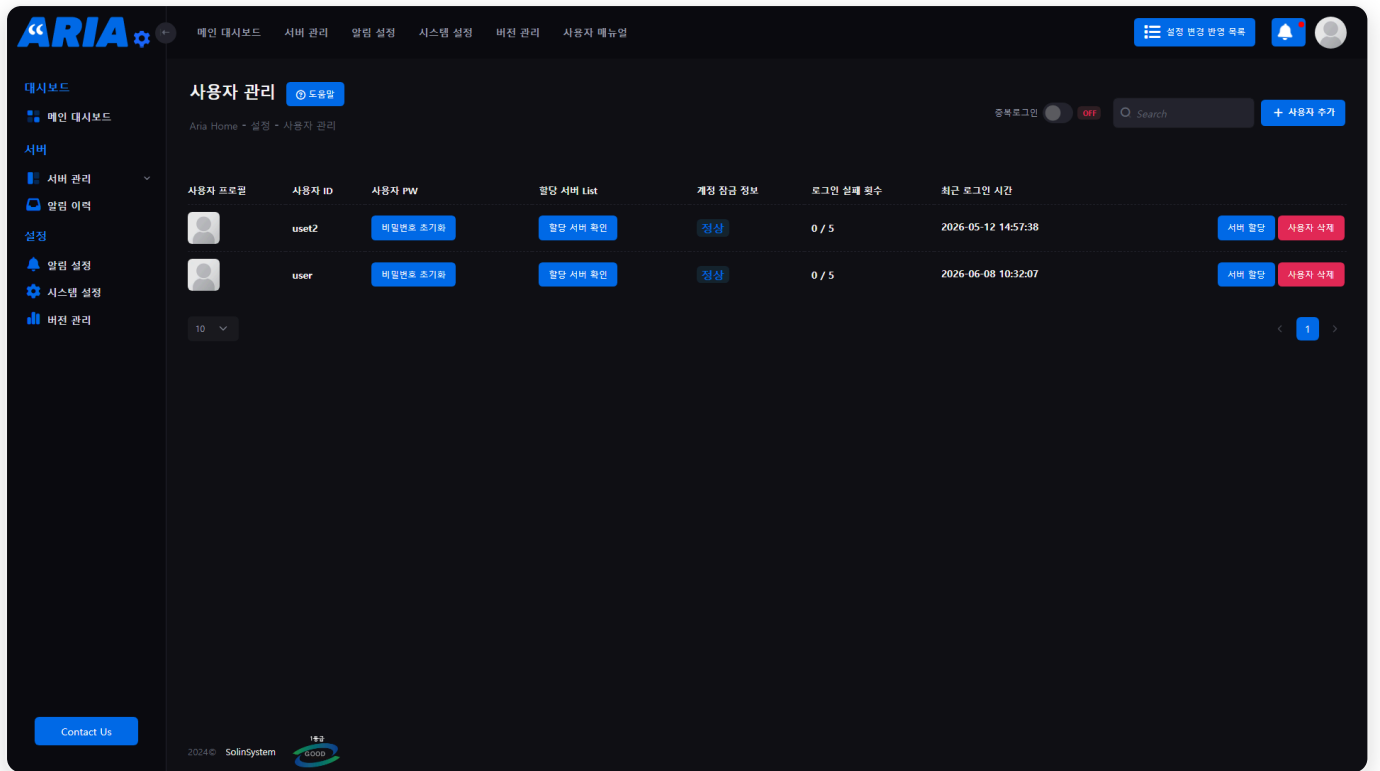
- **정기 변경:** 보안 정책에 따라 3~6개월마다 비밀번호를 변경하세요.
- **관리자 계정:** admin 계정의 비밀번호는 [사용자 관리](#)와 별도로 이 화면에서 직접 관리합니다.

4.4 사용자 관리

경로: 좌측 사이드바 **설정·계정 > 사용자 관리**

ARIA 시스템에 접속할 수 있는 계정(Account)을 통합 관리하는 보안·계정 관리 페이지입니다. 신규 사용자 생성, 비밀번호 초기화, 그리고 **사용자별 접근 가능한 서버 권한 분리**까지 한 화면에서 처리할 수 있습니다.

4.4.1 화면



4.4.2 1. 화면 구성

영역	설명
검색	사용자 ID 또는 이름으로 실시간 필터링
사용자 추가 버튼	신규 계정 생성
사용자 목록 테이블	등록된 계정 정보 및 관리 버튼

테이블 컬럼

컬럼	설명
ID	로그인에 사용하는 계정 식별자
이름	사용자 표시 이름
권한	관리자 / 일반 사용자 구분
최근 로그인	마지막 접속 일시

컬럼	설명
관리 버튼	비밀번호 초기화, 할당 서버 확인, 서버 할당, 삭제

2. 신규 사용자 추가

1. **사용자 추가** 버튼을 클릭합니다.
2. 계정 정보를 입력합니다.

항목	필수	설명
사용자 ID	✓	로그인에 사용할 고유 식별자
이름	✓	화면에 표시될 이름
초기 비밀번호	✓	최초 로그인 시 사용할 임시 비밀번호
권한	✓	관리자(모든 기능) / 일반 사용자(할당된 서버만)

1. **저장**을 클릭하여 계정을 생성합니다.

참고: 일반 사용자로 생성된 계정은 서버 할당 전까지 [메인 대시보드](#)에서 어떤 서버도 조회할 수 없습니다.

4.4.3 3. 서버 할당 (권한 분리)

일반 사용자가 조회·제어할 수 있는 서버를 지정합니다.

1. 서버를 할당할 사용자의 **서버 할당** 버튼을 클릭합니다.
2. 팝업에서 해당 사용자에게 접근 권한을 부여할 서버를 선택합니다.
3. **저장**을 클릭합니다.

할당 서버 확인

할당 서버 확인 버튼으로 특정 사용자에게 현재 어떤 서버가 할당되어 있는지 확인할 수 있습니다.

보안 팁: 정기적으로 할당 서버 현황을 감사하여 퇴사자나 담당 변경자의 권한이 적절히 조정되었는지 확인하세요.

4.4.4 4. 비밀번호 초기화

사용자가 비밀번호를 분실했거나 보안 정책에 따라 강제 변경이 필요할 때 사용합니다.

1. 대상 사용자의 **비밀번호 초기화** 버튼을 클릭합니다.
2. 확인 모달에서 초기화를 확정합니다.
3. 임시 비밀번호가 설정되며, 사용자가 다음 로그인 시 새 비밀번호를 설정하도록 안내합니다.

4.4.5 5. 사용자 삭제

퇴사자나 더 이상 시스템 접근이 필요 없는 계정을 영구적으로 삭제합니다.

1. 삭제할 사용자의 **삭제** 버튼을 클릭합니다.
2. 확인 모달에서 대상을 검토한 후 삭제를 확정합니다.

주의: 삭제된 계정은 복구할 수 없습니다. 반드시 삭제 전 대상 계정을 확인하세요.

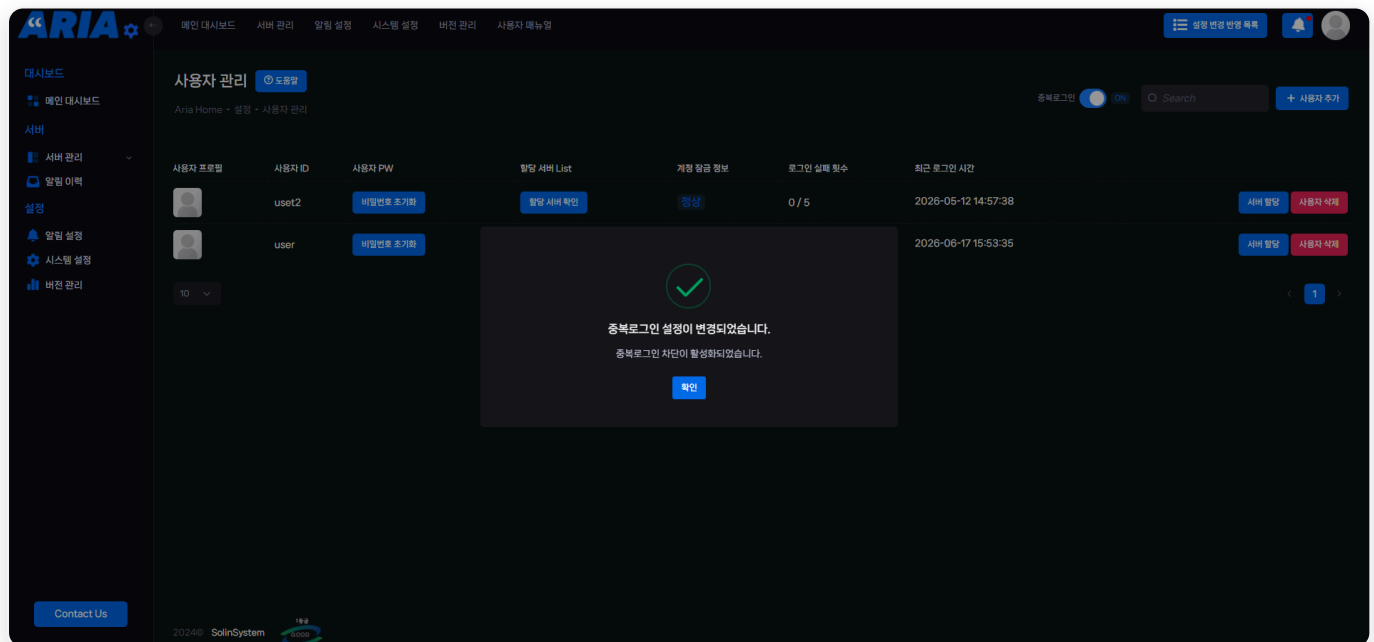
4.4.6 6. 중복 로그인 방지 설정

ARIA 시스템은 다중 동시 접속을 차단하여 계정 도용을 예방하고 보안을 강화할 수 있는 **중복 로그인 차단** 설정을 제공합니다.

6-1. 중복 로그인 차단 활성화

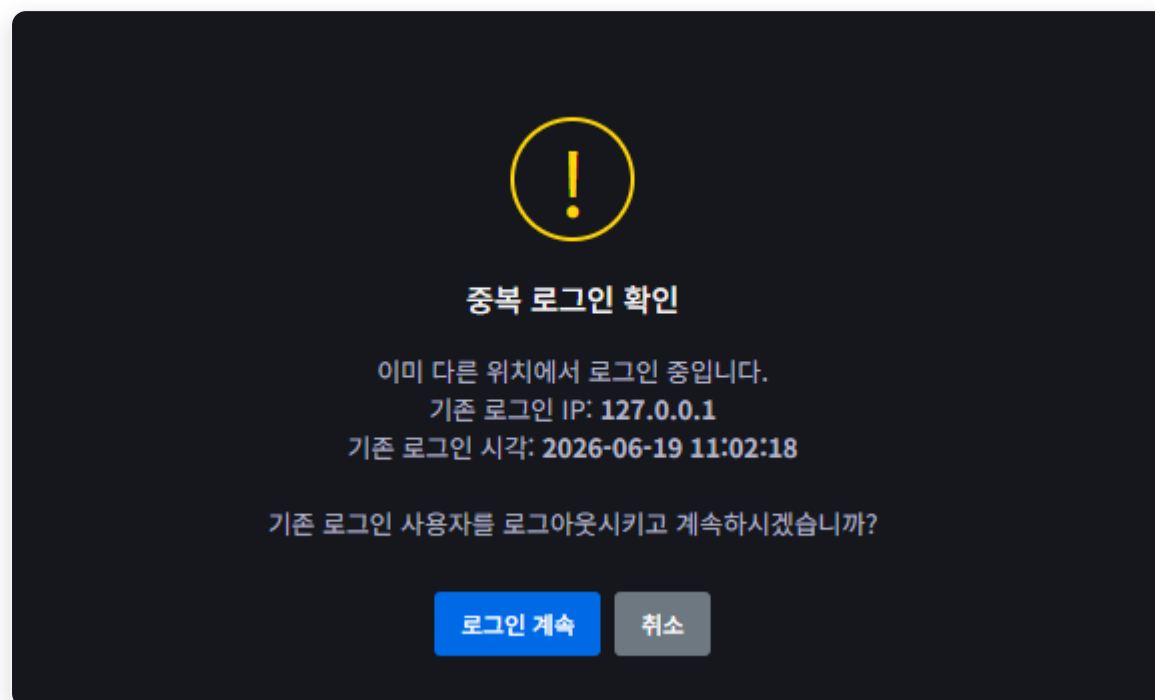
사용자 관리 화면 상단의 **중복 로그인 차단** 토글 스위치를 통해 활성화할 수 있습니다.

- **토글 ON:** 한 계정당 동시에 1개의 세션(클라이언트)만 로그인을 유지하도록 제한합니다.
- **토글 OFF:** 여러 브라우저나 디바이스에서 동일 계정의 동시 중복 접속을 허용합니다.



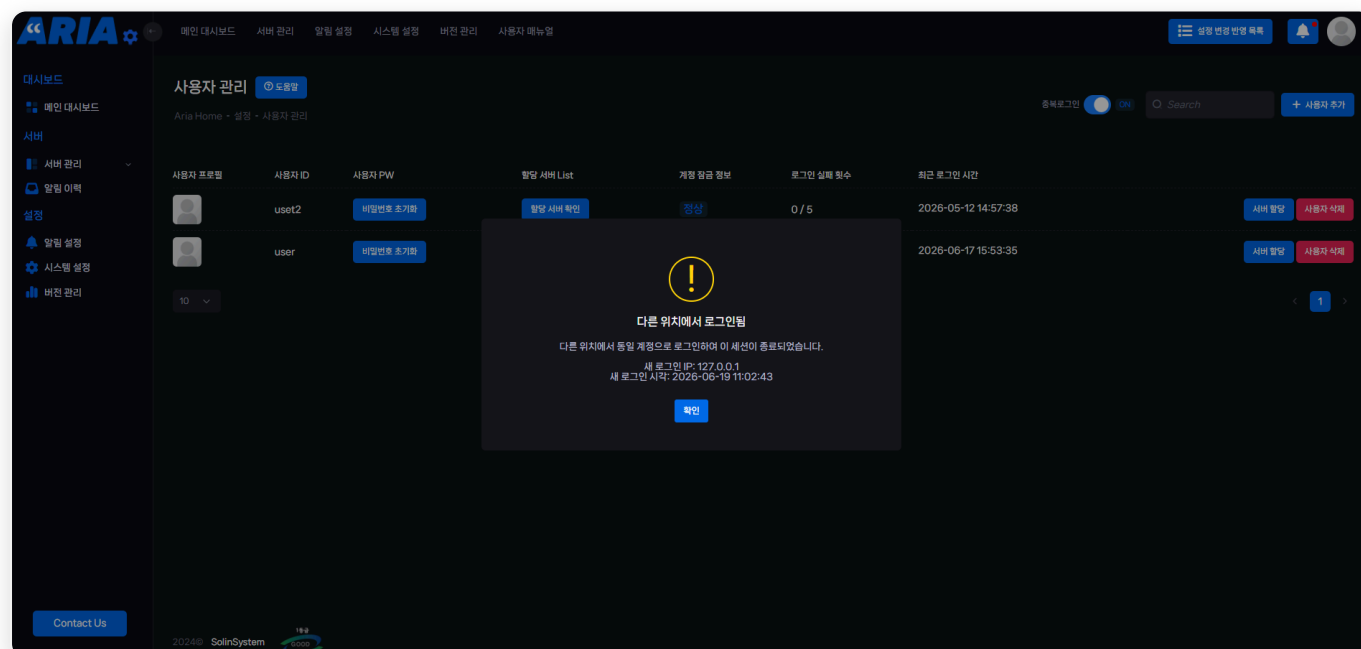
6-2. 중복 로그인 감지 및 접속 시도

중복 로그인 차단이 켜진 상태에서 다른 환경(기기 또는 타 브라우저)으로 동일한 계정 로그인을 시도하는 경우, 로그인 창에 기존 접속 중인 세션 정보(로그인 IP, 접속 시각)를 띄우며 기존 접속을 종료하고 새 로그인을 계속 진행할 것인지 묻는 경고 팝업이 노출됩니다.



6-3. 기존 세션 만료 처리

새로운 로그인을 강행하여 완료하면 기존 클라이언트 세션은 즉시 만료되며, 이전 화면에는 "다른 기기에서 로그인하여 세션이 만료되었습니다" 경고와 함께 새 로그인 IP·접속 시각이 안내됩니다.



4.4.7 💡 활용 팁

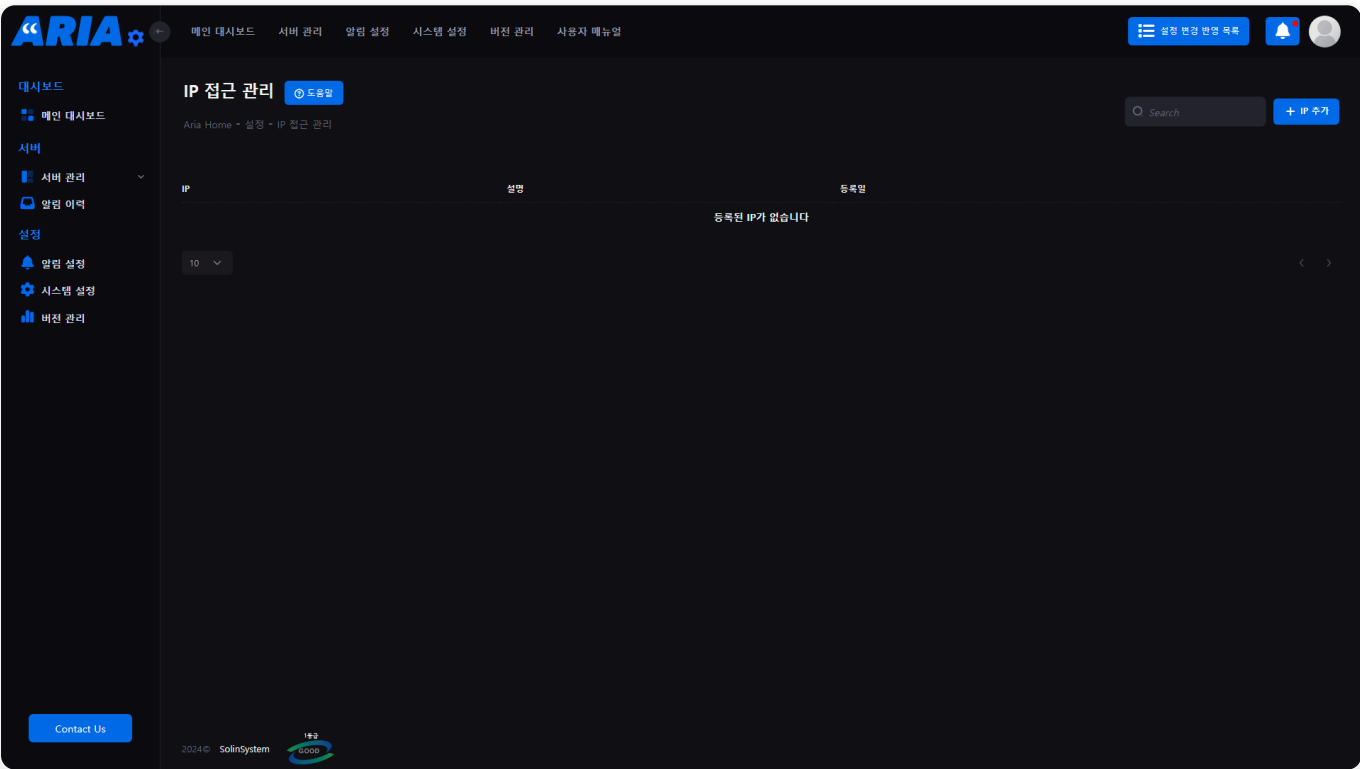
- **최소 권한 원칙:** 일반 사용자에게는 담당 서버만 할당하고, 관리자 권한은 필요한 계정에만 부여하세요.
- **IP 접근 제어:** [IP 접근 관리](#)와 함께 사용하면 보안을 강화할 수 있습니다.
- **프로필 관리:** 사용자 본인은 [프로필](#) 화면에서 비밀번호를 직접 변경할 수 있습니다.

4.5 IP 설정 (IP 차단 관리)

경로: 좌측 사이드바 [설정·계정](#) > [IP 접근 관리](#)

ARIA 서버로 접속을 시도하는 비인가 사용자의 접근을 차단하기 위해 특정 IP 주소나 대역을 등록하고 관리하는 보안 설정 페이지입니다.

4.5.1 화면



4.5.2 1. IP 차단 목록 조회

IP 설정 페이지 진입 시 현재 등록되어 있는 차단 IP 목록 테이블을 조회할 수 있습니다.

컬럼	설명
차단 IP 주소	차단 대상 IP 주소
설명	차단 사유나 등록 시 작성한 메모

컬럼	설명
등록일	차단 IP가 시스템에 등록된 일시
상태	현재 차단 규칙의 활성화 여부(ON/OFF 토글)
관리	등록된 차단 IP를 수정하거나 목록에서 삭제

4.5.3 2. 차단 IP 추가

새로운 비인가 접근 시도 IP나 보안 위협이 감지된 IP를 차단 목록에 추가합니다.

1. 화면 우측 상단의 **IP 추가** 버튼을 클릭하여 등록 모달 창을 엽니다.
2. 모달 창 내 입력 항목을 기입합니다.
3. **IP 주소**: 차단할 대상의 고유 IP 주소를 정확히 입력합니다. (예: 192.168.0.150)
4. **설명**: 차단 목적 또는 세부 메모를 기입하여 타 관리자가 파악할 수 있도록 돕습니다.
5. **저장**을 클릭하여 설정을 완료하면 즉시 해당 IP의 매니저 서버 접속이 차단됩니다.

4.5.4 3. 차단 IP 관리 및 해제

- **임시 차단 해제 (토글)**: 테이블 목록의 상태 토글 스위치를 클릭하여 차단 규칙을 즉시 활성화(ON)하거나 비활성화(OFF)할 수 있습니다.
- **차단 IP 삭제**: 불필요하거나 오등록된 차단 IP는 관리 컬럼의 **삭제(휴지통 아이콘)** 버튼을 눌러 목록에서 완전히 제거하여 접속을 다시 허용할 수 있습니다.

4.5.5 💡 활용 팁

- **오등록 주의**: 관리자 IP를 실수로 차단하면 본인 접속이 불가해질 수 있으므로 등록 전 IP를 재확인하세요.
- **계정 보안 연계**: [사용자 관리](#)의 중복 로그인 차단과 함께 운용하면 다층 보안을 구성할 수 있습니다.

5 서버 관리

5.1 서버 목록

경로: 좌측 사이드바 서버 관리 > 서버 목록

ARIA에 등록된 모든 Tomcat 에이전트 서버 목록을 조회하고 관리하는 화면입니다. 서버를 새로 추가하거나, 기존 서버의 정보를 수정·삭제하고, 개별 서버의 에이전트 관리 화면으로 이동할 수 있습니다.

관리자 전용 기능: 서버 추가·수정·삭제는 관리자(admin) 계정만 수행할 수 있습니다. 일반 사용자는 할당된 서버 목록만 조회할 수 있습니다.

5.1.1 화면

The screenshot displays the ARIA Server Management interface. The top navigation bar includes links for Home, Server Management, Agent Settings, System Settings, Backup Management, and User Management. The left sidebar shows the navigation menu with options like Dashboard, Server Management, and Settings. The main content area is titled '서버 목록' (Server List) and shows a summary of 9 total servers, with 4 alive and 5 dead. Below this, there is a table listing individual servers with columns for Status, Agent Name, IP, Port, OS, Tomcat Version, Embedded Tomcat, and Manager Exist. Each row includes a '관리창 이동' (Move to Management) button.

STATUS	AGENT NAME	IP	PORT	OS	TOMCAT VERSION	EMBEDDED TOMCAT	MANAGER EXIST
Alive	Importance1 test2	192.168.0.153	6900	Rocky Linux 9.4	11.0.22	✗	✓
Alive	Importance11 test	192.168.0.153	6904	Rocky Linux 9.4	11.0.22	✗	✓
Dead	test2 기본그룹	1.1.1.1	2	LINUX	-	✗	✗
Dead	test3 기본그룹	1.1.1.1	3	LINUX	-	✗	✗
Dead	test4 기본그룹	1.1.1.1	4	LINUX	-	✗	✗
Dead	test5 기본그룹	1.1.1.1	5	LINUX	-	✗	✗
Dead	test6 기본그룹	1.1.1.1	6	LINUX	-	✗	✗

5.1.2 1. 서버 추가

서버 추가 버튼을 클릭하면 새 Tomcat 에이전트 서버를 등록하는 입력 화면으로 이동합니다. 에이전트가 설치된 서버의 IP·포트·Catalina 경로·시작/정지 스크립트를 입력해야 합니다.

자세한 입력 방법은 [서버 추가](#) 문서를 참고하세요.

2. 등록 대기 서버

등록 대기 서버는 ARIA 에이전트가 연동된 Tomcat이 기동되면 ARIA가 자동으로 감지하고, 관리자가 **한 번에 여러 대를 일괄 등록**할 수 있는 기능입니다. 서버를 한 대씩 수동 입력하는 **서버 추가** 방식보다 빠르게 대량의 신규 인스턴스를 온보딩할 때 특히 유용합니다.

관리자 전용: 등록 대기 서버 버튼과 일괄 등록은 **관리자(admin)** 계정에서만 사용할 수 있습니다.

동작 원리

1. ARIA 에이전트가 설치된 Tomcat이 기동되면, 에이전트는 모니터링·로그 등의 데이터를 ARIA로 전송합니다.
2. ARIA는 전송된 **에이전트 IP와 포트**가 **이미 등록된 서버**인지 확인합니다.
3. **미등록** 인스턴스는 수집 데이터를 처리하지 않고, 대신 **등록 대기 목록**(서버 메모리)에 IP·포트·Catalina Base 경로·OS 정보를 축적합니다.
4. 관리자가 서버 목록에서 **등록 대기 서버** 버튼을 열어 내용을 확인·수정한 뒤 **선택 항목 등록**으로 **서버 목록에 등록**합니다.

에이전트가 함께 보내는 주요 정보는 다음과 같습니다.

항목	설명
에이전트 IP / 에이전트 포트	에이전트가 바인딩된 IP와 통신 포트
Catalina Base 경로	Tomcat 인스턴스의 Catalina Base 경로
OS 분류	Linux 또는 Windows
OS 상세 정보	OS 상세 문자열 (예: Rocky Linux 9.4 (Blue Onyx)) — 등록 시 함께 저장

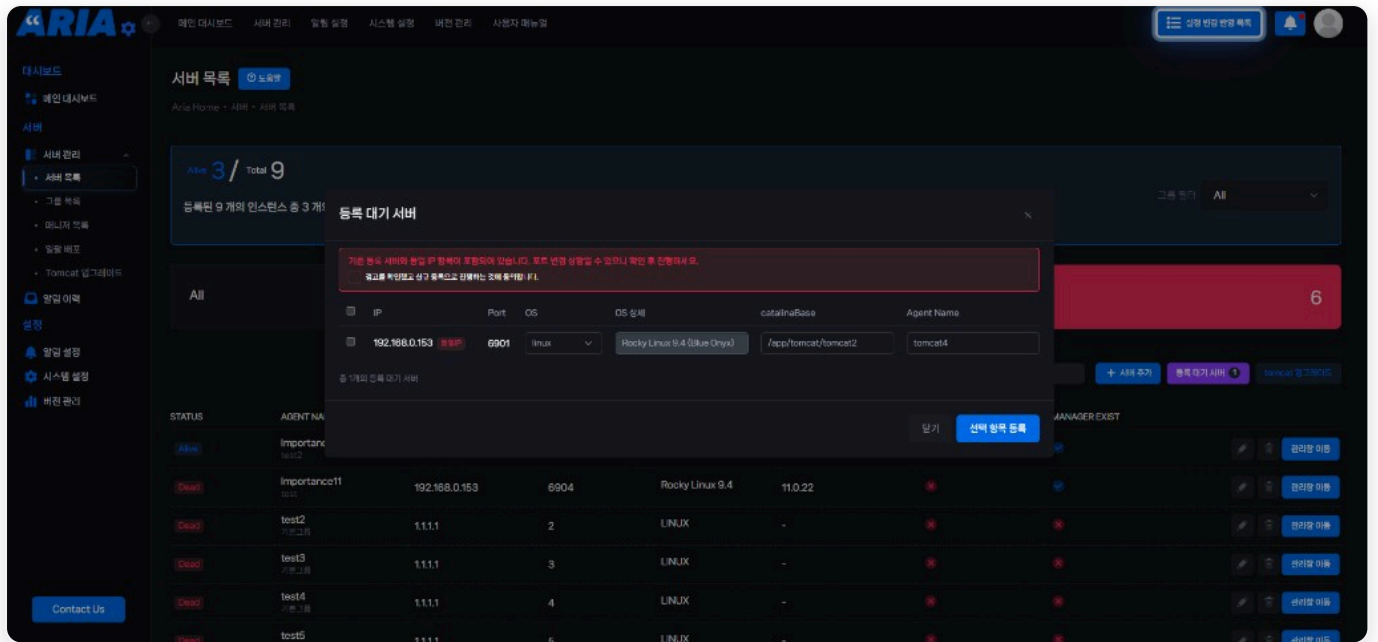
참고: 등록 대기 목록은 ARIA 서버 **메모리**에만 유지됩니다. ARIA 서버를 재기동하면 목록이 초기화되며, 이후 미등록 에이전트가 다시 데이터를 내면 그때부터 다시 쌓입니다.

버튼 표시

서버 추가 버튼 옆에 **등록 대기 서버** 버튼이 표시됩니다.

- 배지 숫자는 현재 등록 대기 중인 서버 수입니다.
- 대기 건이 **1건 이상**이면 버튼이 강조 색상으로 표시되어 확인이 필요함을 알립니다.
- 대기 건이 없어도 버튼은 열 수 있으며, 이 경우 "등록 대기 서버가 없습니다." 메시지가 표시됩니다.

등록 대기 서버 모달



모달을 열면 감지된 서버가 표 형태로 나타납니다. 상단 **전체 선택** 체크박스로 여러 행을 한꺼번에 선택할 수 있습니다.

컬럼	설명
선택	일괄 등록 대상 여부
IP	감지된 에이전트 IP. 기존 등록 서버와 IP가 같으면 동일IP 배지 표시
Port	감지된 에이전트 포트
OS	linux / windows 선택 (에이전트가 보낸 값을 기본으로 채움)
OS 상세	에이전트가 수집한 OS 상세 정보 (읽기 전용)
catalinaBase	Tomcat Catalina Base 경로 (필요 시 수정 가능)
Agent Name	서버 목록에 표시될 서버 이름 (기본값 tomcat1, tomcat2 ... 자동 부여)

행 옆에 **manager 없음** 배지가 보이면, 해당 IP에 ARIA Manager가 아직 등록되지 않은 상태입니다. 이 경우 서버 등록 전에 Manager 등록이 필요합니다.

일괄 등록 절차

1. 등록 대기 서버 버튼을 클릭해 모달을 엽니다.
2. 등록할 행을 체크하고, **Agent Name**, **catalinaBase**, **OS** 값을 필요에 맞게 수정합니다.
3. 선택 항목 등록 버튼을 클릭합니다.

등록 시 ARIA는 아래 항목을 자동으로 채웁니다.

- **시작/정지 스크립트:** Catalina Base 경로와 OS에 따라 startup.sh / shutdown.sh(Linux) 또는 startup.bat / shutdown.bat(Windows) 경로 생성
- **OS 상세:** 에이전트가 수집한 OS 상세 정보 저장
- **등록 후처리:** 기존 **서버 추가**와 동일하게 캐시 갱신, Watchdog 등록, Manager 동기화 수행

등록이 끝나면 성공·실패 건수와 Manager 동기화 결과가 표시되고, 화면이 새로고침됩니다.

동일 IP 경고

목록에 **동일IP** 배지가 붙은 항목이 있으면 상단에 경고가 표시됩니다.

■ 기존 등록 서버와 동일 IP 항목이 포함되어 있습니다. 포트 변경 상황일 수 있으니 확인 후 진행하세요.

같은 물리 서버에서 **포트만 다른 Tomcat 인스턴스**를 새로 띄운 경우에 자주 나타납니다. 의도한 신규 인스턴스가 맞다면 경고 확인 체크박스에 동의한 뒤 등록을 진행하세요.

Manager가 없는 경우

선택한 서버 IP에 Manager가 등록되어 있지 않으면, 등록 전에 **Manager 추가** 안내가 표시됩니다.

1. 대상 IP를 확인합니다.
2. Manager Port를 입력합니다. (기본값 **6810**)
3. Manager 상태 확인 후 등록을 완료합니다.
4. Manager 추가가 끝나면 등록 대기 목록이 갱신되며, 이후 일괄 등록을 이어갈 수 있습니다.

등록 실패 시 확인 사항

상황	조치
Agent Name 중복	다른 이름으로 변경 후 다시 등록
동일 IP 경고 미확인	경고 확인 체크박스 선택 후 재시도
Manager 미등록	위 Manager 추가 절차 수행
등록 대기 목록에서 사라짐	에이전트가 일정 시간 동안 통신하지 않으면 목록에서 제거될 수 있음. Tomcat·에이전트 기동 상태 확인 후 모달을 다시 열기

■ **활용 팁:** 신규 서버를 여러 대 동시에 기동한 뒤 **등록 대기 서버** 모달에서 한 번에 선택·등록하면, 수동 **서버 추가**보다 훨씬 빠르게 온보딩할 수 있습니다.

5.1.3 3. Tomcat 업그레이드

Tomcat 업그레이드 버튼을 클릭하면 **Tomcat 업그레이드** 화면으로 이동합니다. ARIA를 통해 Tomcat 바이너리를 원격으로 업그레이드할 수 있으며, 업그레이드 전 현재 버전과 대상 버전을 반드시 확인하세요.

주의: 업그레이드 시 Tomcat 서비스가 일시 중단됩니다. 운영 서버에서는 점검 시간을 확보한 후 진행하세요.

5.1.4 4. 서버 정보 수정 및 삭제

수정

목록에서 서버를 선택한 후 **수정** 버튼을 클릭하면 해당 서버의 기본 정보(이름, IP, 포트, 스크립트 경로 등)를 수정하는 화면으로 이동합니다.

삭제

삭제 버튼을 클릭하면 확인 모달이 표시됩니다. 삭제를 확정하면 해당 서버가 ARIA 목록에서 제거됩니다. 삭제해도 실제 서버나 에이전트 프로그램에는 영향을 주지 않습니다.

주의: 삭제된 서버는 복구할 수 없습니다. 대상 서버를 반드시 확인한 후 삭제하세요.

5.1.5 5. 에이전트 관리 화면으로 이동

관리창 이동 버튼을 클릭하면 선택한 서버의 **에이전트 대시보드**로 이동합니다. 에이전트 관리 화면에서는 Tomcat 시작/정지, 설정 편집, 모니터링, 배포 등 서버별 세부 작업을 수행할 수 있습니다.

기능	연관 문서
시작/정지	시작/정지
모니터링	모니터링
WAR 배포	어플리케이션 배포
로그 조회	로그 뷰어
설정 편집	server.xml 등
설정 반영	설정 변경 반영

5.1.6 💡 활용 팁

- **등록 대기 서버 우선 확인:** 신규 Tomcat을 기동한 뒤에는 서버 목록의 **등록 대기 서버** 배지를 먼저 확인하세요. 감지된 인스턴스를 일괄 등록하면 수동 입력보다 빠릅니다.
- **그룹별 관리:** 서버가 많을 경우 [그룹 목록](#)에서 운영·개발·테스트 환경별로 그룹을 나누어 관리하세요.
- **일괄 작업:** 여러 서버에 동일 WAR를 배포할 때는 [일괄 배포](#) 기능을 활용하세요.

- **Dead 상태 서버:** 목록에서 Dead로 표시된 서버는 에이전트 연결을 먼저 확인한 후 **서버 추가** 절차의 연결 확인 항목을 참고하세요.

5.2 그룹 목록

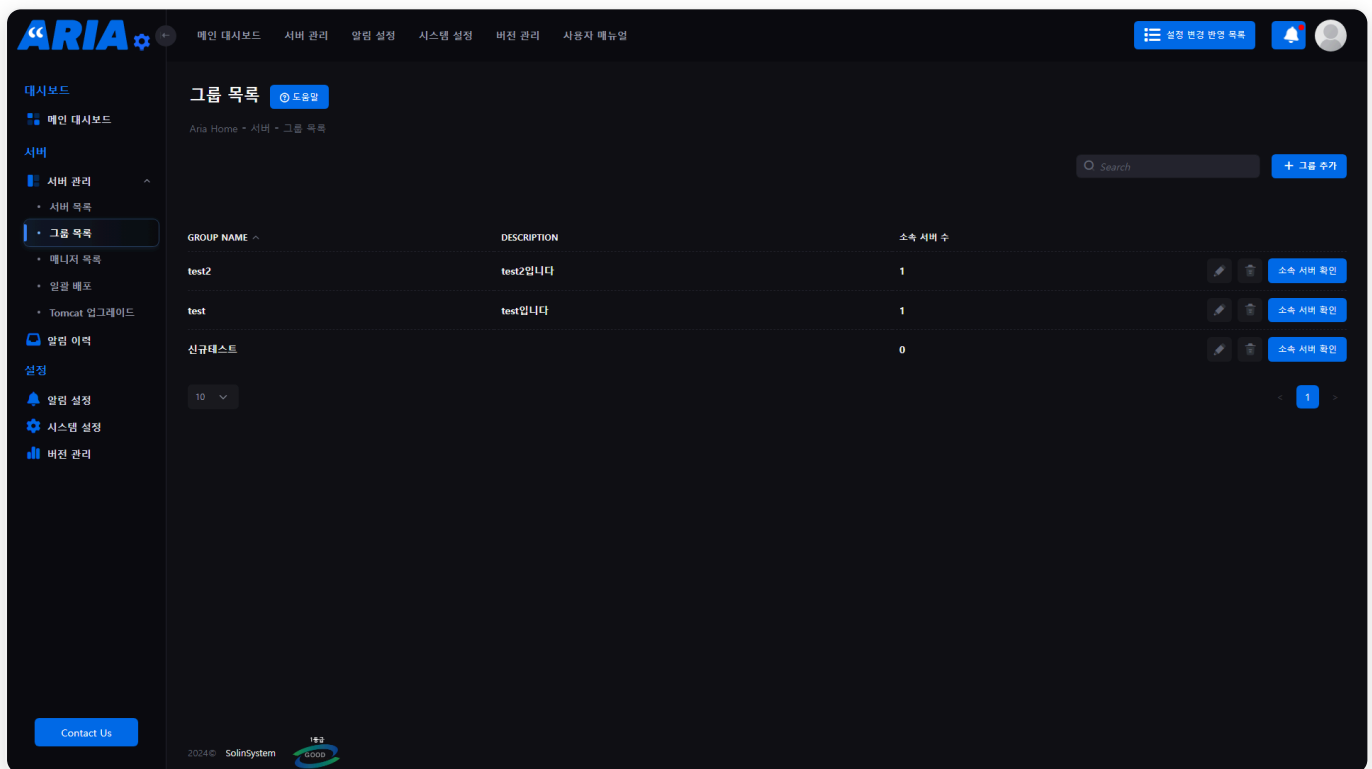
경로: 좌측 사이드바 **서버 관리** > **그룹 목록**

서버 그룹은 여러 Tomcat 서버를 논리적으로 묶어 관리하는 단위입니다. 예를 들어, 운영계·개발계·테스트계처럼 환경별로 그룹을 나누거나, 서비스 A팀·B팀처럼 조직 단위로 묶을 수 있습니다.

그룹을 사용하면 **메인 대시보드**에서 서버들을 그룹 단위로 정렬·필터링하여 더 쉽게 현황을 파악할 수 있습니다.

권한 안내: 그룹 추가·수정·삭제와 소속 서버 모달에서의 **서버 수정·삭제**는 **관리자(admin)** 계정만 수행할 수 있습니다. 일반 사용자는 **소속 서버 확인**으로 자신에게 할당된 서버만 조회할 수 있습니다.

5.2.1 화면



5.2.2 1. 그룹 추가

그룹 추가 버튼을 클릭하면 새 서버 그룹을 생성하는 입력 모달이 열립니다.

- 그룹 이름과 설명을 입력한 후 저장하면 그룹이 생성됩니다.
- 그룹 생성 후 **소속 서버 확인** 기능으로 서버를 그룹에 지정할 수 있습니다.

2. 그룹 정보 수정

목록에서 그룹을 선택한 후 수정 버튼을 클릭하면 그룹 이름과 설명을 변경할 수 있습니다.

5.2.3 3. 그룹 삭제

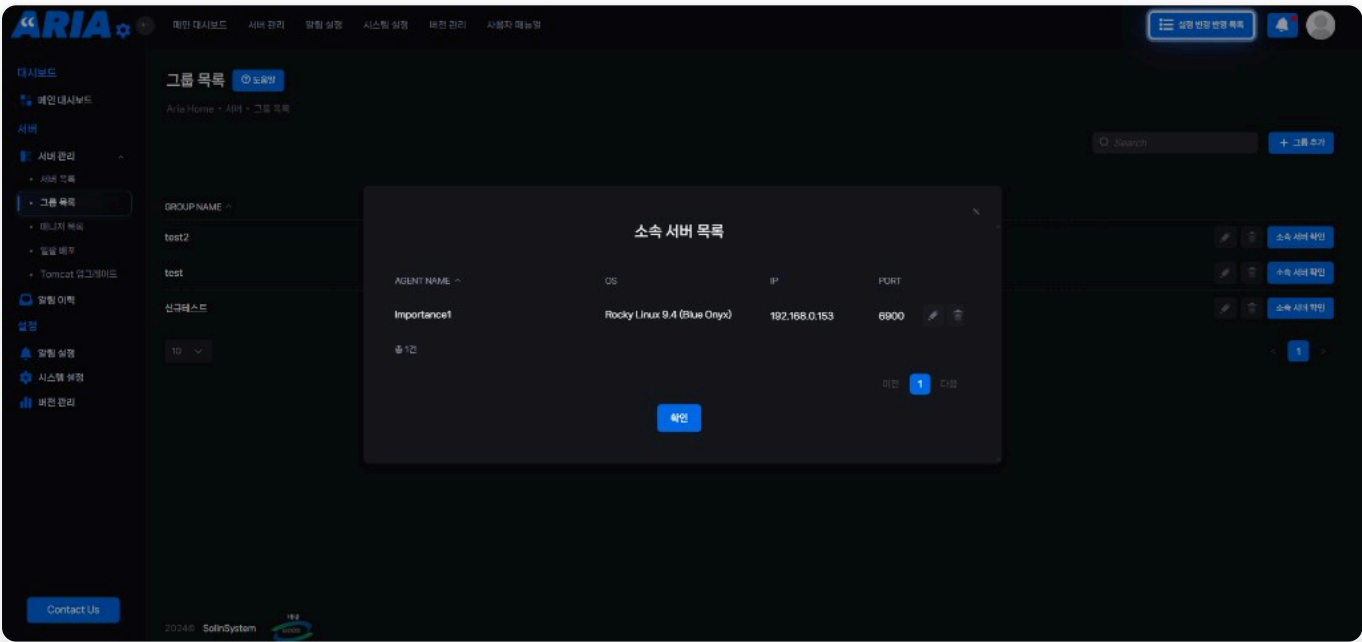
삭제 버튼을 클릭하면 확인 모달이 표시됩니다.

- 그룹을 삭제해도 해당 그룹에 속해 있던 서버는 삭제되지 않으며, 서버는 미분류(그룹 없음) 상태로 유지됩니다.

주의: 삭제된 그룹은 복구할 수 없습니다.

5.2.4 4. 소속 서버 확인

각 그룹 행의 소속 서버 확인 버튼을 클릭하면 소속 서버 목록 모달이 열립니다. 해당 그룹에 배정된 Tomcat 서버를 한눈에 확인할 수 있습니다.



모달 구성

컬럼	설명
Agent Name	서버 이름
OS	OS 상세 정보 우선 표시 (예: Rocky Linux 9.4 (Blue Onyx)). 없으면 Linux / Windows 분류 표시
IP	에이전트 IP
Port	에이전트 포트

목록은 Agent Name 기준으로 정렬되며, 서버가 많을 경우 페이지당 5건씩 나뉘어 표시됩니다. 소속 서버가 없으면 "소속된 서버가 없습니다." 메시지가 나타납니다.

조회 권한

계정	조회 범위
관리자(admin)	해당 그룹에 속한 모든 서버
일반 사용자	해당 그룹에 속하면서 본인에게 할당된 서버만

관리자 전용: 서버 수정·삭제

관리자로 로그인한 경우, 각 행 오른쪽에 **수정**(연필) · **삭제**(휴지통) 버튼이 표시됩니다. 일반 사용자에게는 이 버튼 열이 보이지 않으며, 목록 조회만 가능합니다.

서버 수정

수정 버튼을 클릭하면 **서버 정보 수정** 화면(/servers/updateServer)으로 이동합니다. 여기서 서버 이름, IP, 포트, **소속 그룹**, Catalina 경로 등을 변경할 수 있습니다.

- 다른 그룹으로 옮기려면 **그룹** 필드를 변경한 뒤 저장하세요.
- 미분류 상태로 두려면 그룹 선택을 해제(지정되지 않음)하면 됩니다.

서버 삭제

삭제 버튼을 클릭하면 확인 모달이 표시됩니다. 확인 시 해당 서버가 **ARIA 서버 목록에서 완전히 제거**됩니다.

주의: 이 삭제는 그룹에서만 빼는 것이 아니라 **서버 등록 자체를 삭제**하는 동작입니다. 실제 Tomcat·에이전트 프로세스에는 영향을 주지 않지만, ARIA 목록에서는 복구할 수 없습니다.

5.2.5 5. 그룹 생성 후 서버 배정 절차

1. **그룹 추가** 버튼을 클릭하여 그룹 이름과 설명을 입력하고 저장합니다.
2. **서버 목록**에서 대상 서버의 **수정**을 클릭하거나, 그룹의 **소속 서버 확인** 모달(관리자)에서 **수정** 버튼으로 서버 정보 화면으로 이동합니다.
3. **그룹** 필드에서 방금 만든 그룹을 선택하고 저장합니다.
4. 그룹 목록에서 **소속 서버 확인**을 열어 서버가 반영되었는지 확인합니다.
5. **메인 대시보드**에서 그룹별로 서버가 정렬되어 표시되는 것을 확인합니다.

팁: **등록 대기 서버**로 신규 서버를 일괄 등록한 뒤, 서버 수정 화면에서 그룹을 지정하면 대량 온보딩 후 그룹 배정까지 빠르게 마무리할 수 있습니다.

5.2.6 💡 활용 팁

- **소속 서버 모달에서 바로 관리:** 관리자는 그룹별 서버 현황을 보면서 모달에서 곧바로 수정·삭제할 수 있어, **서버 목록**으로 이동하지 않고도 빠르게 점검할 수 있습니다.
- **일괄 배포 연계:** **일괄 배포** 시 그룹 단위로 대상 서버를 선택하면 운영·개발 환경을 분리하여 안전하게 배포할 수 있습니다.
- **권한 관리:** **사용자 관리**에서 일반 사용자에게 그룹 단위로 서버 접근 권한을 부여하는 것을 검토하세요.

5.3 매니저 목록

경로: 좌측 사이드바 **서버 관리 > 매니저 목록**

ARIA Manager는 대상 호스트에 설치되는 제어 프로세스입니다. Tomcat 내장 Manager 앱이 아니라, ARIA Server가 Tomcat **시작/정지, 설정 파일 반영, 패치·업그레이드** 등을 원격으로 수행할 때 사용하는 중간 계층입니다.

한 호스트(IP)에 Tomcat 인스턴스가 여러 개 있어도 **ARIA Manager 하나**가 `agent.xml` 등을 통해 해당 호스트의 에이전트들을 관리합니다.

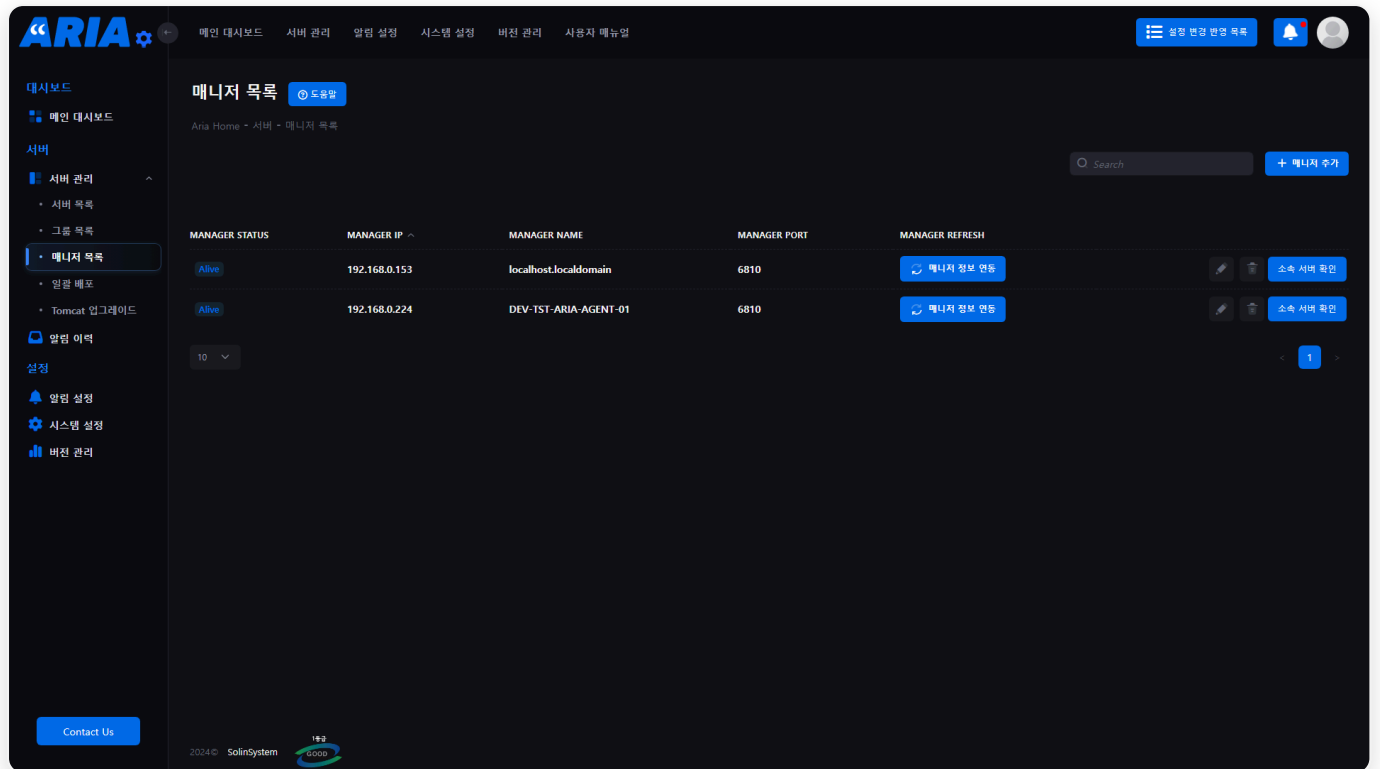
준필수 구성 요소: Manager가 등록·기동되지 않은 서버 IP에서는 아래 기능을 사용할 수 없습니다.

기능	Manager 없을 때
Tomcat 시작/정지	불가
설정 파일 변경·반영	불가
Tomcat 업그레이드	불가
ARIA 업데이트(패치)	불가

모니터링·로그 조회 등 **수집(Agent) 기반 기능**은 Manager 없이도 동작할 수 있으나, 운영 제어가 필요한 환경에서는 Manager 등록을 전제로 구성하는 것을 권장합니다.

관리자 전용: 매니저 추가·수정·삭제, 매니저 정보 연동은 **관리자(admin)** 계정만 수행할 수 있습니다.

5.3.1 화면



목록에는 등록된 Manager의 **상태(Alive/Dead)**, **IP**, **이름(호스트명)**, **포트**가 표시됩니다.

5.3.2 1. 매니저 추가

매니저 추가 버튼을 클릭하면 등록 모달이 열립니다.

항목	설명
Manager IP	Manager가 설치된 호스트 IP (옥텟 4칸 입력)
Manager Port	Manager HTTP 포트 (기본값 6810)

등록 절차는 다음과 같습니다.

1. IP·포트를 입력합니다.
2. 완료 클릭 시 ARIA Server가 `http://{IP}:{Port}/MANAGER_HEALTH` 로 **기동 여부를 확인**합니다.
3. 응답이 정상이면 호스트명을 **Manager 이름**으로 자동 저장하고 **Manager 목록에 등록**합니다.
4. 동일 IP가 이미 등록되어 있으면 "이미 존재하는 Manager IP입니다." 오류가 표시됩니다.

사전 조건: 대상 호스트에 ARIA Manager가 설치되어 있고, ARIA Server에서 해당 IP·포트로 HTTP 접근이 가능해야 합니다.

2. 매니저 정보 연동

매니저 정보 연동 버튼은 Manager 상태가 **Alive**일 때만 사용할 수 있습니다.

클릭하면 ARIA Server가 해당 Manager와 **서버(agent) 정보를 다시 동기화**합니다. Manager에 연결된 Tomcat 인스턴스 목록 · agent.xml 등이 최신 상태로 맞춰집니다.

- 서버를 새로 등록했거나 Manager 쪽 인벤토리가 어긋났을 때 사용합니다.
- **등록 대기 서버** 일괄 등록 후에도 자동으로 Manager 동기화가 수행됩니다.
- Manager가 **Dead**이면 버튼이 비활성화됩니다. Manager 프로세스 기동·방화벽(기본 6810)을 먼저 확인하세요.

5.3.3 3. 매니저 정보 수정

수정(연필) 버튼을 클릭하면 포트 변경 모달이 열립니다.

항목	수정 가능
Manager IP	읽기 전용
Manager Name (호스트명)	읽기 전용
Manager Port	수정 가능 (1~65535)

포트 변경 후 저장하면 연결 상태를 다시 확인합니다. 연결에 실패하면 "매니저 서버 연결에 실패했습니다." 메시지가 표시됩니다.

5.3.4 4. 매니저 삭제

삭제(휴지통) 버튼을 클릭하면 확인 모달이 표시됩니다.

확인 시 **Manager 목록**에서 해당 Manager 등록 정보가 제거됩니다. 이미 등록된 Tomcat **서버 목록** 항목 자체는 삭제되지 않지만, 해당 IP에서 **시작/정지·설정 반영·업그레이드** 등 Manager 의존 기능은 사용할 수 없게 됩니다.

5.3.5 5. 소속 서버 확인

소속 서버 확인 버튼을 클릭하면 **소속 서버 목록** 모달이 열립니다. 해당 **Manager IP**와 **동일한 에이전트 IP**를 가진 Tomcat 서버가 표시됩니다. 즉, 이 Manager가 제어하는 호스트 위의 에이전트 목록을 확인하는 화면입니다.

소속 서버 목록

SERVER NAME	SERVER OS	SERVER PORT
Importance1	Rocky Linux 9.4	6900
Importance11	Rocky Linux 9.4	6904
tomcat2	Rocky Linux 9.4	6903
tomcat3	Rocky Linux 9.4	6902

확인

모달 구성

컬럼	설명
Server Name	서버 이름
Server OS	OS 상세 정보 우선 표시 (예: Rocky Linux 9.4). 없으면 Linux / Windows 분류
Server port	에이전트 포트

목록은 Server Name 기준으로 정렬됩니다. 소속 서버가 없으면 "소속된 서버가 없습니다."가 표시됩니다.

이 모달은 **조회 전용**입니다. 서버 수정·삭제는 [서버 목록](#) 또는 [그룹 목록](#)의 소속 서버 모달(관리자)에서 수행합니다.

활용: [서버 목록](#)의 **MANAGER EXIST** 열과 함께 보면, 특정 IP에 Manager가 등록되어 있고 실제로 어떤 Tomcat들이 묶여 있는지 빠르게 점검할 수 있습니다.

5.3.6 💡 활용 팁

- **설치 순서:** 호스트에 ARIA Manager 설치 → **매니저 목록**에서 IP 등록 → **서버 목록**에서 Tomcat(에이전트) 등록 → 필요 시 **매니저 정보 연동**으로 동기화.

- **등록 대기 서버와 연계:** [등록 대기 서버](#) 일괄 등록 시 Manager가 없는 IP는 등록 전에 Manager 추가를 안내합니다.
- **Dead 상태:** Alive가 아닌 Manager는 연동·제어가 불가합니다. Manager 프로세스, 포트(6810), 방화벽, ARIA Server→Manager 네트워크 경로를 확인하세요.
- **역할 구분:** **ARIA Data Agent**는 모니터링 수집, **ARIA Manager**는 Tomcat **제어**를 담당합니다. 자세한 아키텍처는 [시작하기](#) 문서를 참고하세요.

5.4 Tomcat 업그레이드

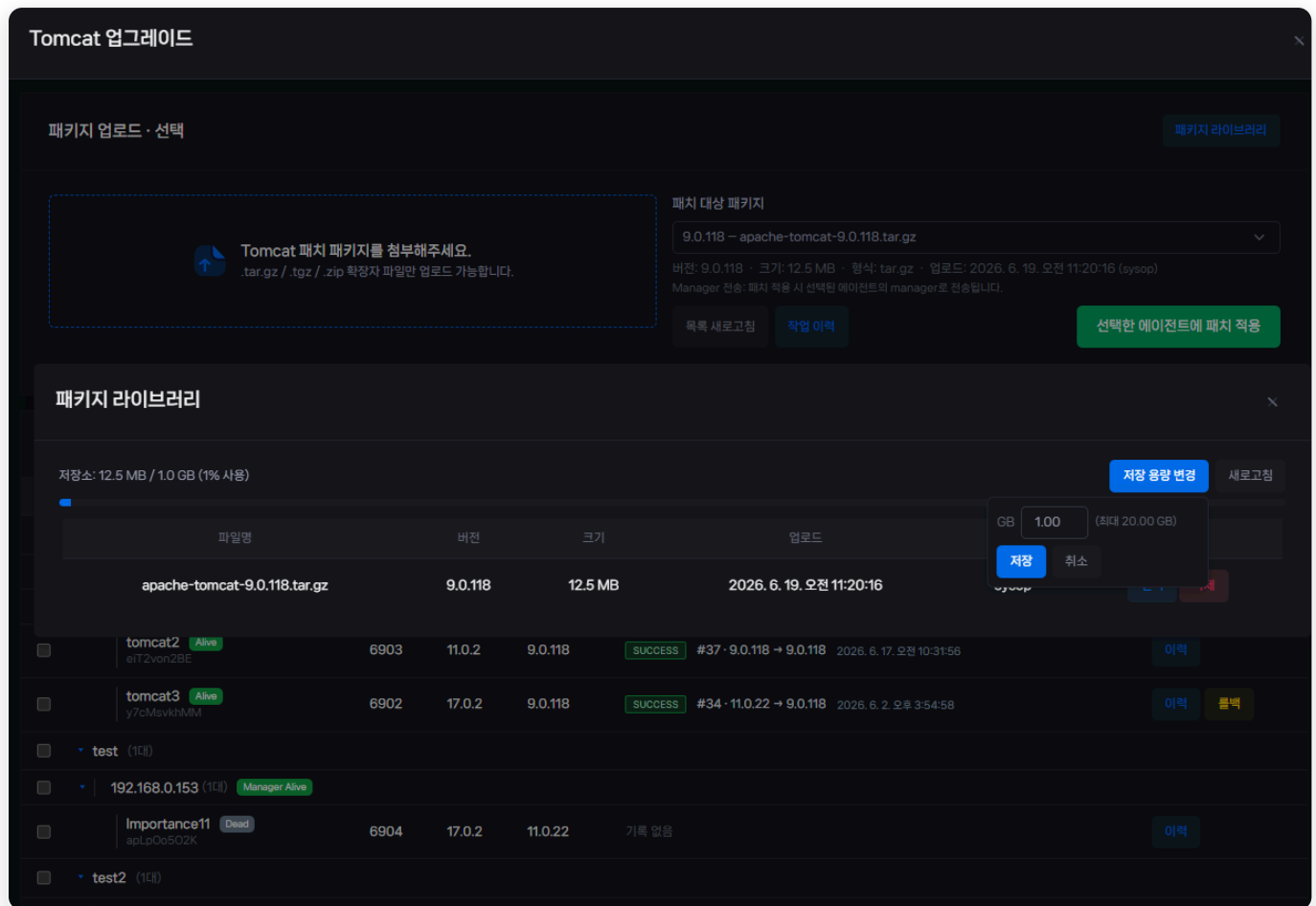
■ 경로: 좌측 사이드바 **서버 관리 > Tomcat 업그레이드** 메뉴 클릭 (또는 서버 목록의 **Tomcat 업그레이드** 버튼 클릭)

다수의 서버(에이전트)에 설치된 Tomcat 엔진을 원격으로 간편하게 업그레이드하는 전용 관리 기능입니다. 일일이 서버에 SSH로 접속하여 바이너리를 교체할 필요 없이, 웹 콘솔에서 패치 패키지를 업로드하여 대상 에이전트들에 일괄 전송하고 버전을 업데이트 할 수 있습니다.

5.4.1 1. 패치 패키지 등록 및 라이브러리 관리

Tomcat 업그레이드에 사용할 패키지 파일을 업로드하고 관리합니다.

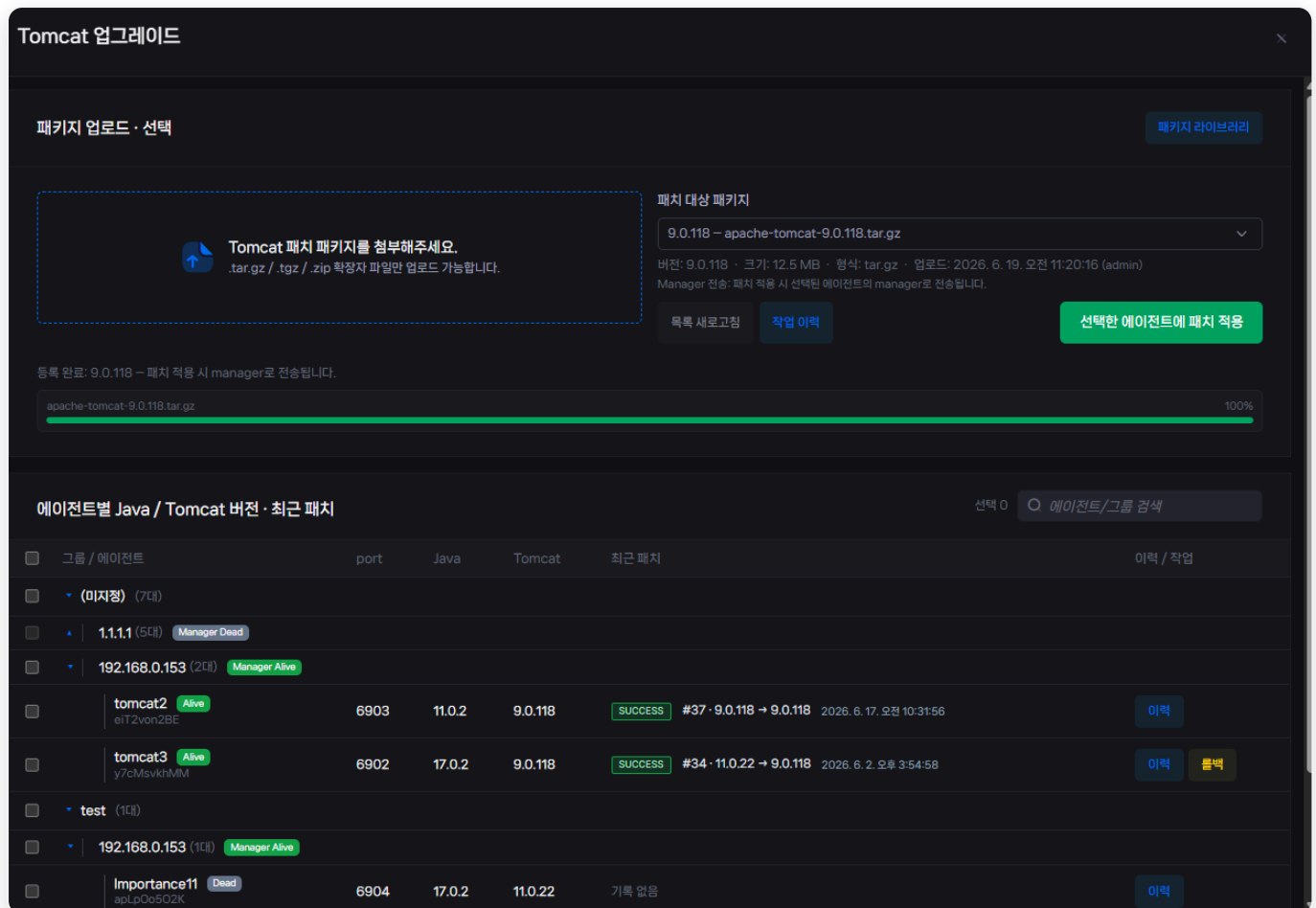
- **지원 확장자:** `.tar.gz`, `.tgz`, `.zip` 포맷의 아카이브 파일을 지원합니다.
- **패키지 영구 저장 및 재사용:** 업로드된 패키지 파일은 ARIA Server 내부에 저장되어 영구 보관되며, 추후 다른 서버를 패치할 때 **패키지 라이브러리**에서 선택하여 재사용할 수 있습니다.
- **용량 제한 정책:** 기본적으로 업로드 패키지 파일 용량은 **1GB**로 제한되어 있으나, 설정에 따라 최대 **20GB**까지 용량 제한 확장이 가능합니다.



2. 패키지 파일 업로드 및 진행율 표시

로컬 PC의 패키지 파일을 드래그 앤 드롭하거나 탐색기로 선택하여 업로드할 수 있습니다.

- **실시간 업로드 진행률:** 대용량 패키지 업로드 시, 업로드 영역 하단에 실시간 **Progress Bar**가 표시되어 현재 전송 진행 상태 (%)와 전송량 등을 시각적으로 확인할 수 있습니다.



5.4.2 3. 과거 패치 이력 및 상세 로그 조회

각 에이전트별로 과거에 수행된 패치 히스토리를 확인하여 이력 관리를 투명하게 유지합니다.

- **과거 이력 조회:** 목록에서 대상 에이전트 우측의 **이력** 버튼을 클릭하면 과거에 진행했던 패치 목록(일시, 적용 버전, 상태 등)을 볼 수 있습니다.
- **상세 로그 확인:** 과거 패치 이력의 특정 항목을 선택하면 패치 당시 에이전트에서 실시간으로 출력했던 명령 실행 상세 로그 파일의 전체 텍스트 내용을 팝업 창을 통해 세부적으로 조회할 수 있습니다.

Tomcat 업그레이드

패키지 업로드

패치 이력 - tomcat3

×

라이브러리

버전-파일-상태-jobId 검색

SUCCESS #34 · 11.0.22 → 9.0.118 2026. 6. 2. 오후 3:54:58

실행계정 : admin · 패치 소요시간 : 28초 · 작업 진행률 : 100% · 파일명 : apache-tomcat-9.0.118.tar.gz

로그

SUCCESS #33 · 9.0.118 → 11.0.22 2026. 6. 2. 오후 3:54:05

실행계정 : admin · 패치 소요시간 : 29초 · 작업 진행률 : 100% · 파일명 : apache-tomcat.tar.gz

복백

로그

FAILED #32 · 패치 실패 (작동) · Tomcat stop 실패 2026. 6. 2. 오후 3:52:50

실행계정 : admin · 패치 소요시간 : 3분 32초 · 작업 진행률 : 20% · 파일명 : apache-tomcat.tar.gz

로그

ROLLBACK SUCCESS #31 · 백업복원 (대상 : #26) 2026. 6. 2. 오후 3:42:36

실행계정 : admin · 패치 소요시간 : 13초 · 작업 진행률 : 100% · 파일명 : apache-tomcat-9.0.118.tar.gz

로그

FAILED #30 · 패치 실패 (가동) · 패치 완료 (가동 실패) (11.0.22) · JDK 17 필요 2026. 6. 2. 오후 3:42:12

실행계정 : admin · 패치 소요시간 : 24초 · 작업 진행률 : 90% · 파일명 : apache-tomcat.tar.gz

로그

FAILED #27 · 패치 실패 (작동) · refusing PATCH while Tomcat is still running (stop/forceKill required): catalinaBootstrapPids=[4092327], a... 2026. 6. 2. 오후 3:15:24

실행계정 : admin · 패치 소요시간 : 11초 · 작업 진행률 : 70% · 파일명 : apache-tomcat-9.0.118.tar.gz

로그

DEFAULT #26 · 패치 전 기준점 2026. 6. 2. 오후 3:15:13

실행계정 : admin · 패치 소요시간 : 0ms · 작업 진행률 : 100% · 파일명 : apache-tomcat-9.0.118.tar.gz

로그

총 7건 · 1 / 1 페이지

이전

다음

닫기

복백

에이전트별

그룹 / 에이

· (미지정

· 1.1.

· 192

· t

· t

· test

· 192.168.0.153 (1대)

Manager Alive

Importance11
apLpOo502K

Dead

6904

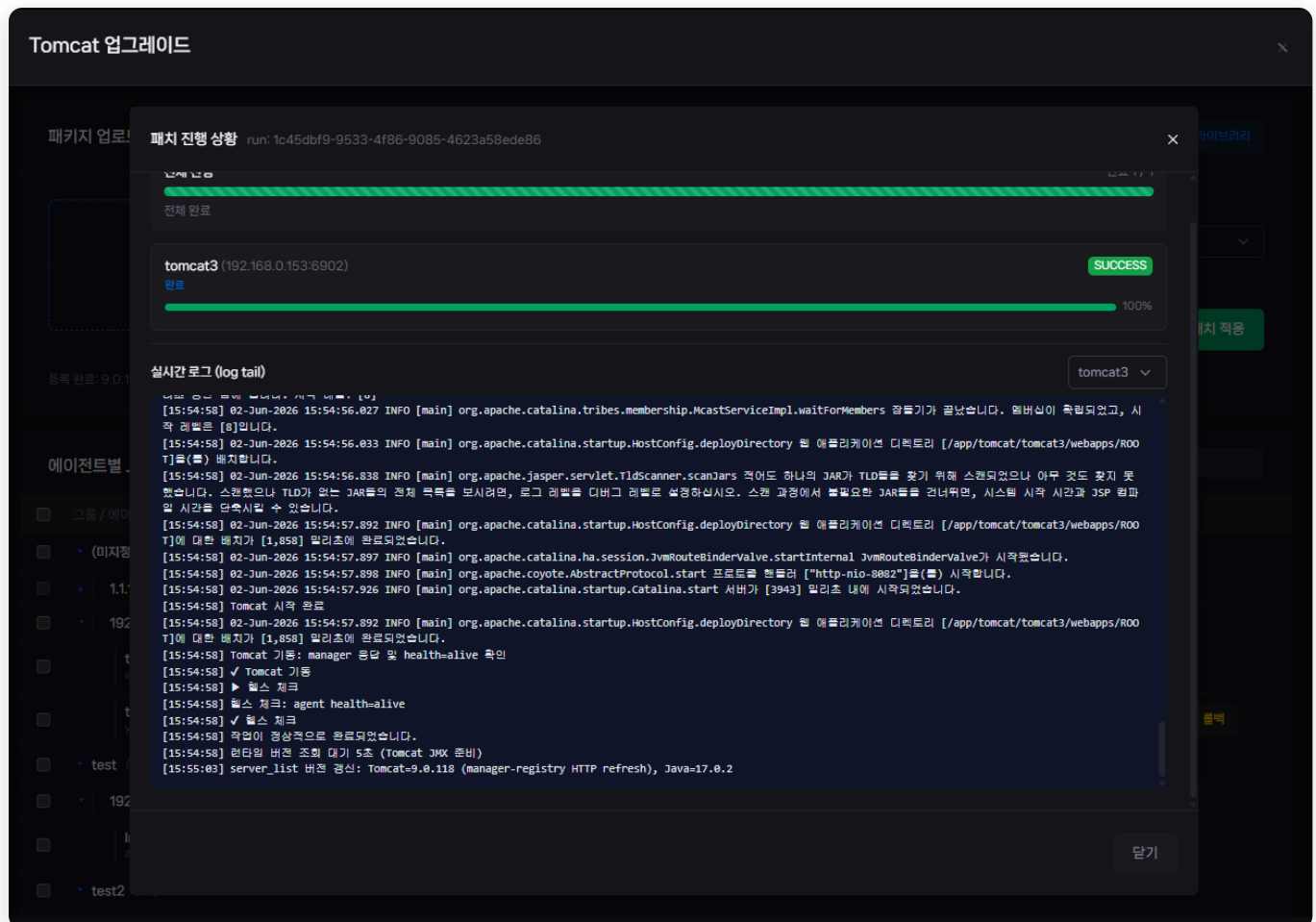
17.0.2

11.0.22

기록 없음

이력

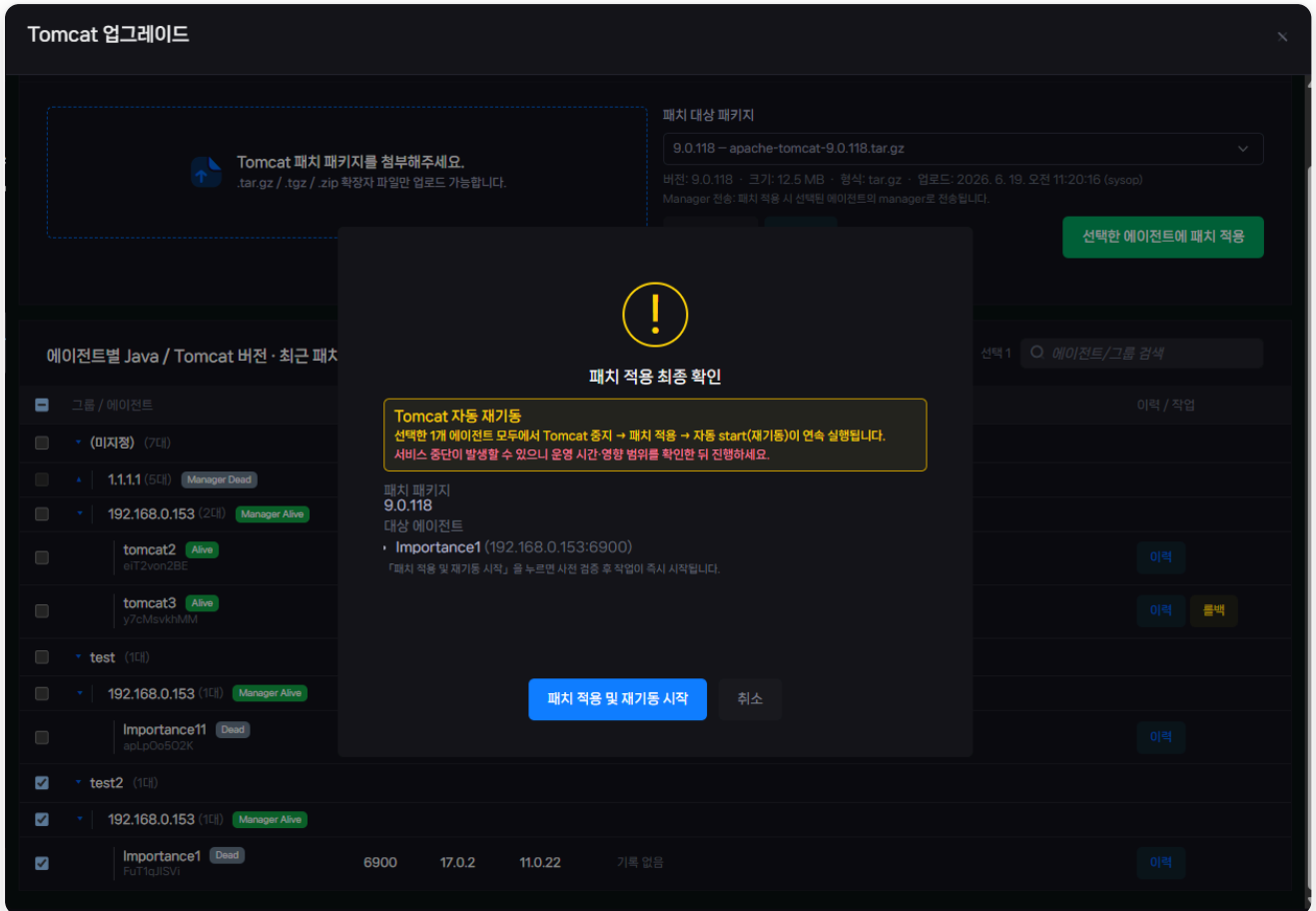
· test2 (1대)



5.4.3 4. 복수 에이전트 지정 및 버전 검증 (패치 실행)

하나 또는 여러 개의 에이전트를 동시에 지정하여 Tomcat 버전을 일괄 업그레이드할 수 있습니다.

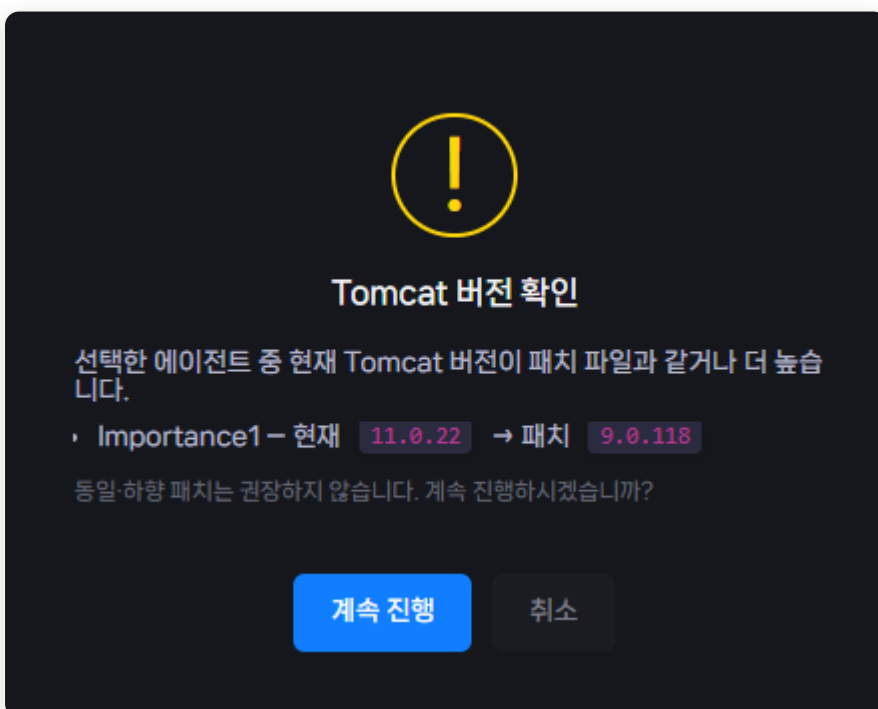
1. 대상 및 패키지 선택: 업그레이드를 적용할 패키지와 타겟 에이전트 서버들을 복수로 선택합니다.



2. 버전 정합성 검증 및 경고:

3. 패치 적용을 누르면 최종 적용할 패키지 버전 정보와 타겟 에이전트 목록이 팝업으로 나타나 최종 승인을 요구합니다.

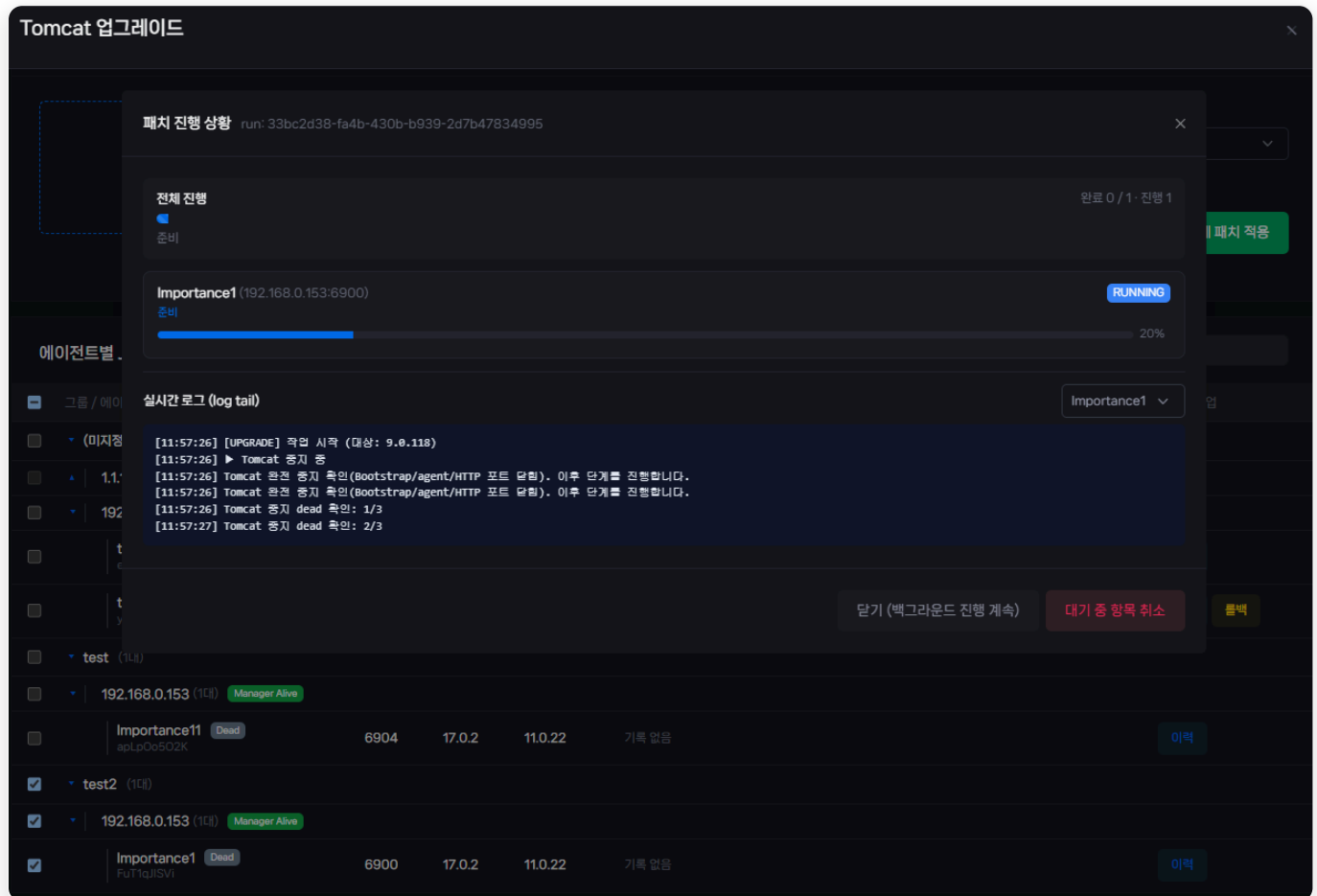
4. **중요 안전 장치:** 만약 선택한 패키지 버전이 대상 에이전트의 **현재 Tomcat 버전과 동일하거나 더 낮은 버전일 경우**, 경고 메시지가 한번 더 출력되어 실수로 인한 다운그레이드를 방지합니다.



5.4.4 5. 실시간 패치 모니터링

패치를 시작하면 에이전트별 패키지 전송, 압축 해제, 파일 교체 및 Tomcat 재시작까지의 전 과정을 실시간으로 모니터링할 수 있습니다.

- **단계별 로그 및 Progress Bar:** 각 에이전트 행마다 패치 진행율이 실시간 **Progress Bar**와 퍼센티지(%)로 나타나며, 하단의 단계별 패치 이력 로그 창을 통해 실시간 커맨드 실행 상황을 모니터링할 수 있습니다.



5.4.5 💡 활용 팁

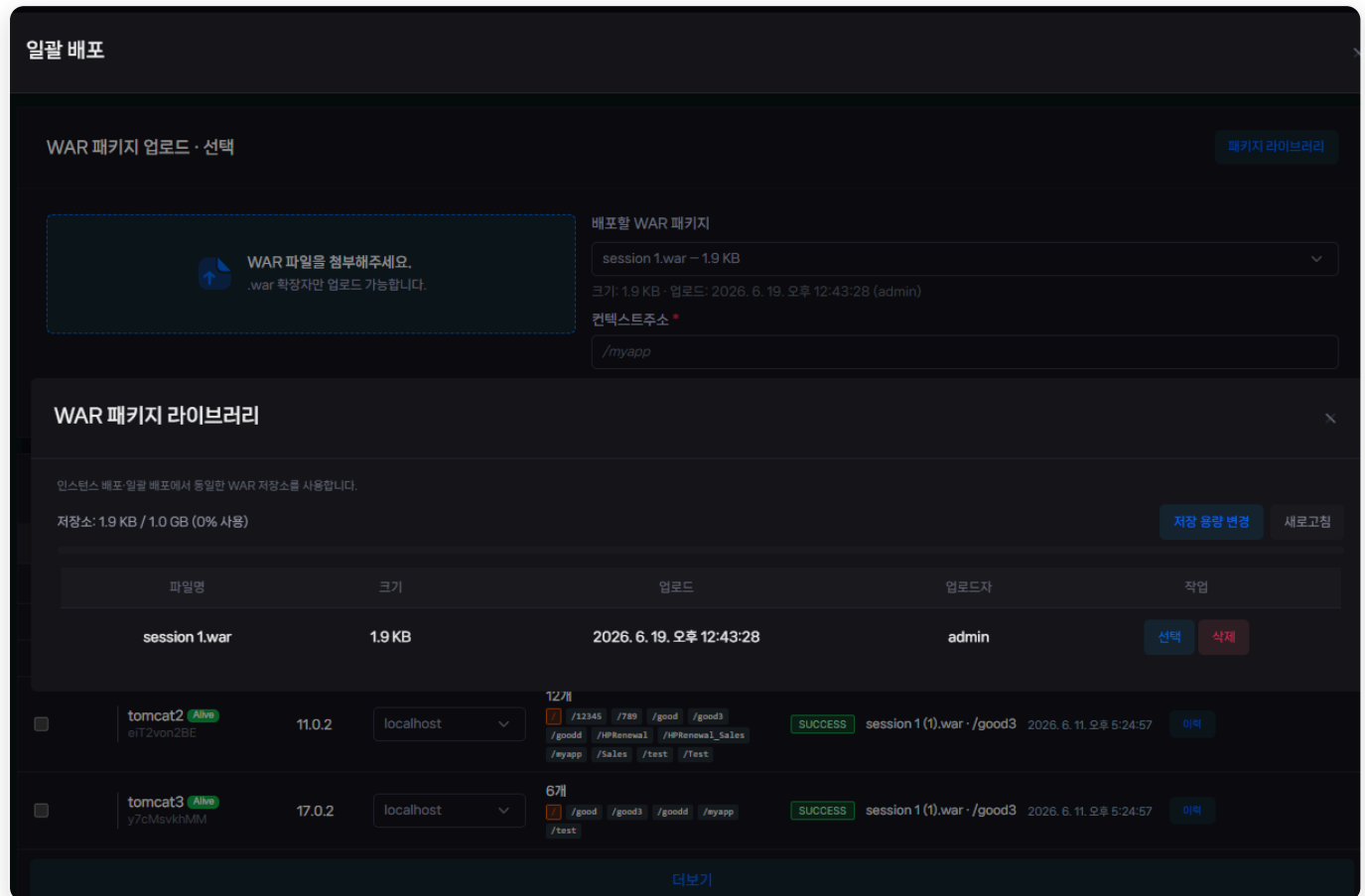
- **버전 호환성 확인:** 패키지 업그레이드 전 에이전트의 **Java 버전**을 조회하여, 새로 적용할 Tomcat 버전이 요구하는 최소 JDK 요구사항을 만족하는지 사전에 교차 검증하세요.
- **서비스 중단 고려:** Tomcat 업그레이드가 진행될 때 Tomcat 웹 프로세스가 자동으로 재시작되므로, 실서비스 운영 서버의 경우 로드 밸런서(L4 등)에서 해당 서버를 일시 제외한 후 순차적으로 순환 패치(Rolling Upgrade)할 것을 강력히 권장합니다.

5.5 일괄 배포

경로: 좌측 사이드바 **서버 관리** > **일괄 배포** 메뉴 클릭

다수의 서버에 동일한 웹 애플리케이션(WAR) 패키지를 일괄 전송하고 배포하는 기능입니다. 전용 모달 창에서 진행되며, 여러 대의 서버에 버전을 업데이트할 때 반복 작업을 줄여줍니다.

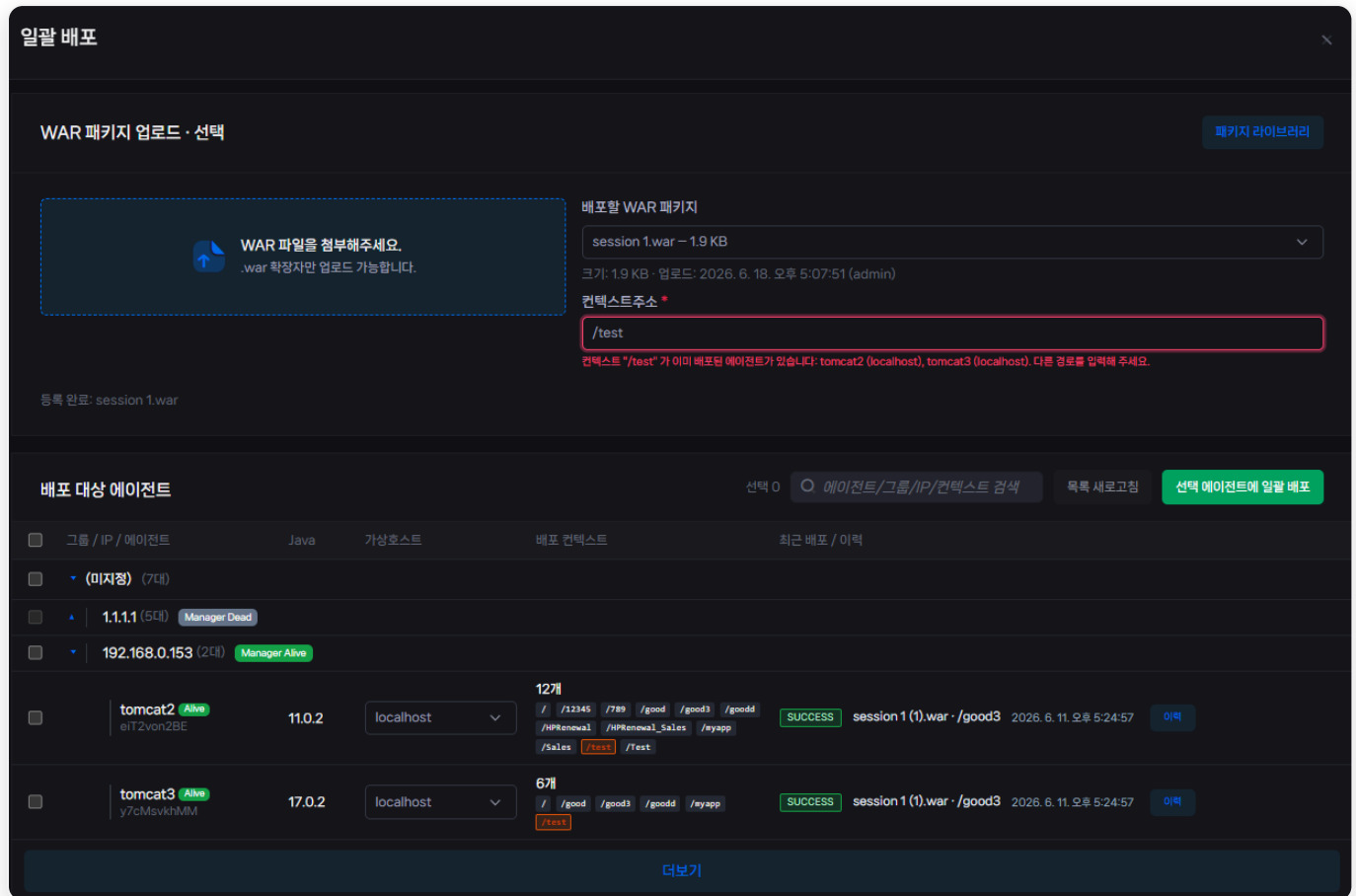
5.5.1 화면



5.5.2 1. 배포 절차

1-1. 배포 패키지(WAR) 및 패키지 라이브러리 선택

1. 배포할 애플리케이션 패키지를 선택하거나 직접 업로드합니다.
2. 패키지 라이브러리 활용: 매번 로컬 PC에서 대용량 WAR 파일을 업로드할 필요 없이, 기존에 업로드되어 ARIA Server 패키지 라이브러리에 등록되어 있는 배포 파일 목록에서 원하는 버전을 클릭하여 즉시 배포용으로 선택할 수 있습니다. 라이브러리 목록을 통해 각 패키지의 버전명, 파일 크기, 수정 일자 등을 파악할 수 있어 효율적이고 일관된 릴리즈 관리가 가능합니다.
3. 서비스가 구동될 Context Path를 확인하고 지정합니다.
4. 중복 검증: 만약 입력한 컨텍스트 주소가 대상 서버의 에이전트에 이미 배포되어 중복되는 경우, 경고 메시지가 표시되며 다른 주소를 입력해야 합니다.



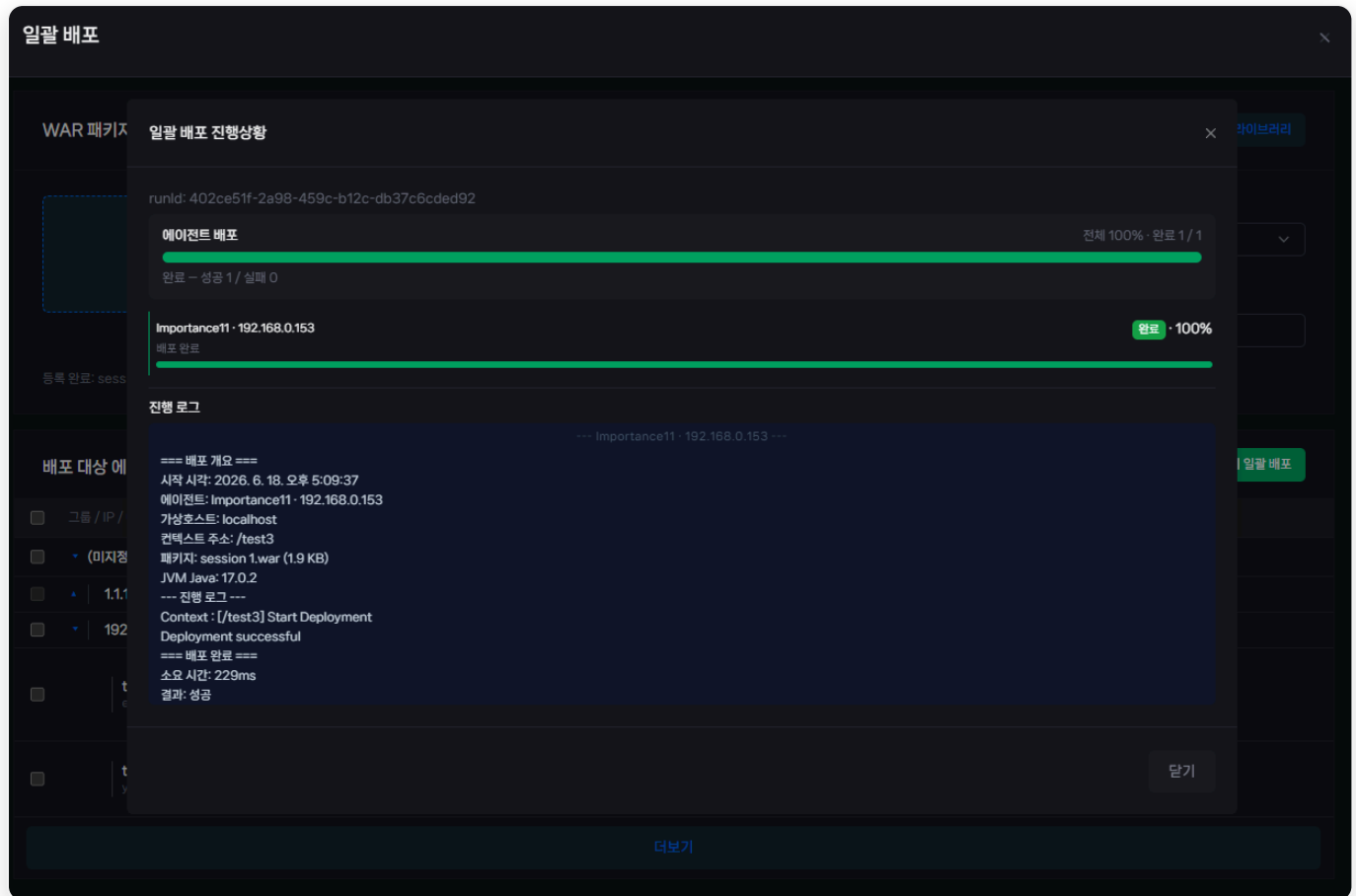
1-2. 배포 대상 에이전트 선택

일괄 배포를 수행할 대상 서버 목록을 지정합니다.

1. 배포 대상 에이전트 목록 패널에서 **그룹**별로 다중 선택하거나, 개별 서버를 선택합니다.
2. 에이전트의 연결 상태(Alive)를 미리 확인하여 배포 실패를 예방합니다.
3. **그룹 목록**에서 미리 환경별 그룹을 구성해 두면 대상 서버 선택이 편리합니다.

1-3. 검증 및 배포 실행

1. 우측 하단의 **배포 검증** 또는 **시작** 버튼을 클릭하여 설정을 검증합니다.
2. 이상이 없으면 실제 배포가 시작되며, **일괄 배포 진행상황** 모달창에서 각 에이전트의 진행률(%)과 실시간 배포 로그를 확인할 수 있습니다.



⚠ 주의: 다수의 운영 서버에 일괄 배포 시 일시적인 서비스 지연이 발생할 수 있으므로, 무중단 배포를 원한다면 그룹을 나누어 순차적으로 배포하세요.

5.5.3 💡 활용 팁

- **무중단 배포 활용:** 전체 서버를 한 번에 배포하기보다, 서버 그룹을 분리(예: 운영-A그룹, 운영-B그룹)하여 한 그룹씩 순차적으로 일괄 배포하면 서비스 무중단 배포(Zero-downtime deployment)를 구현할 수 있습니다.
- **단일 서버 배포:** 한 대의 서버에만 배포할 때는 [어플리케이션 배포](#) 기능을 사용할 수 있습니다.

5.6 설정 파일 가져오기

경로: 에이전트 관리 화면 공통 헤더 우측 **설정 파일 가져오기** 버튼

Tomcat **실서버(원격)** 에 있는 설정 파일 내용을 ARIA Server **작업본**으로 복사하는 기능입니다. 서버에서 직접 수정된 `server.xml` 등을 ARIA에 맞춰 두고, 이후 ARIA 화면에서 편집·이력 관리·반영 절차를 이어갈 때 사용합니다.

대상: 내장형 Tomcat이 아닌 일반 Tomcat 서버에서만 버튼이 표시됩니다.

전제: 해당 호스트에 [ARIA Manager](#)가 등록·기동되어 있어야 Manager 경유로 Tomcat 설정을 조회·가져올 수 있습니다.

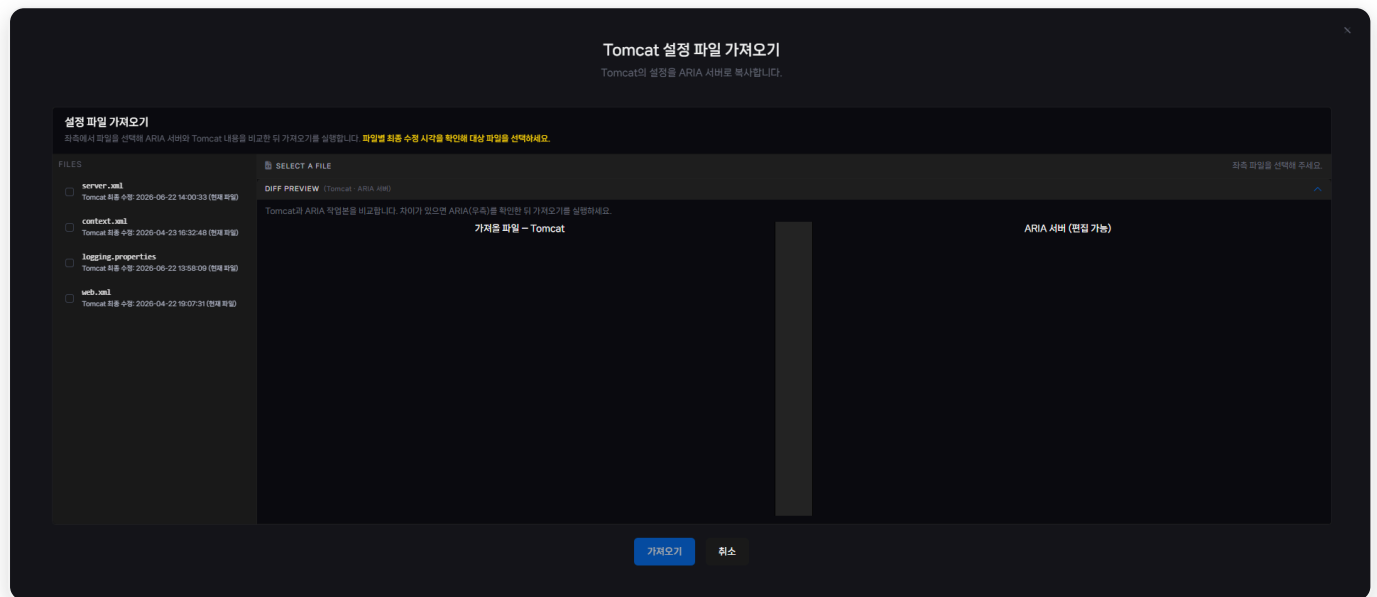
5.6.1 설정 관리 흐름

ARIA의 설정 관련 작업은 보통 아래 순서로 진행합니다.

1. **설정 파일 가져오기** (이 문서) — Tomcat → ARIA 작업본 복사
2. ARIA에서 설정 편집 (`server.xml` 등)
3. **설정 변경 반영** — ARIA 작업본 → Tomcat 적용 및 재기동

가져오기만 수행해도 **Tomcat에는 즉시 반영되지 않습니다**. 반대로, Tomcat에만 반영된 변경을 ARIA에 맞추고 싶을 때 이 기능을 사용합니다.

5.6.2 화면



모달 제목은 **Tomcat 설정 파일 가져오기**이며, 좌측에서 파일을 고르고 우측 **차이 미리보기**로 Tomcat과 ARIA 내용을 비교한 뒤 **가져오기**를 실행합니다.

5.6.3 1. 지원 파일

좌측 **Files** 목록에서 아래 4종 설정 파일을 선택할 수 있습니다.

파일	설명
<code>server.xml</code>	Tomcat 서버 전역 설정
<code>context.xml</code>	Context 공통 설정
<code>logging.properties</code>	로깅 설정
<code>web.xml</code>	웹 애플리케이션 기본 설정

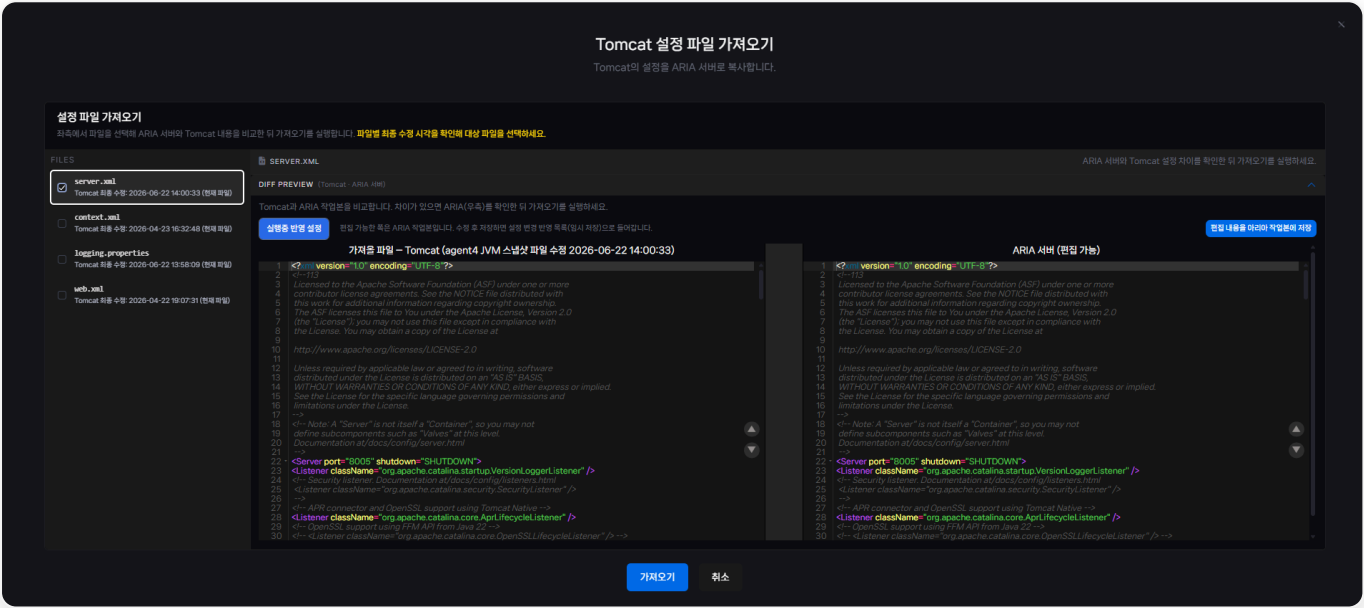
목록을 열면 각 파일의 **Tomcat 최종 수정 시각**이 조회됩니다. 파일별로 시각을 비교해 가져올 대상을 선택하세요.

5.6.4 2. Diff 미리보기

파일을 선택하면 **차이 미리보기 (Tomcat · ARIA 서버)** 영역에 좌우 비교가 표시됩니다.

구분	내용
좌측	가져올 파일 — Tomcat (실서버·실행 중 JVM 기준 조회 우선)
우측	ARIA 서버 (편집 가능) — 현재 ARIA에 저장된 작업본

차이가 있으면 우측 ARIA 작업본을 확인한 뒤 가져오기를 실행합니다. 가져오기는 **Tomcat 내용으로 ARIA 작업본을 덮어쓰는** 동작입니다.

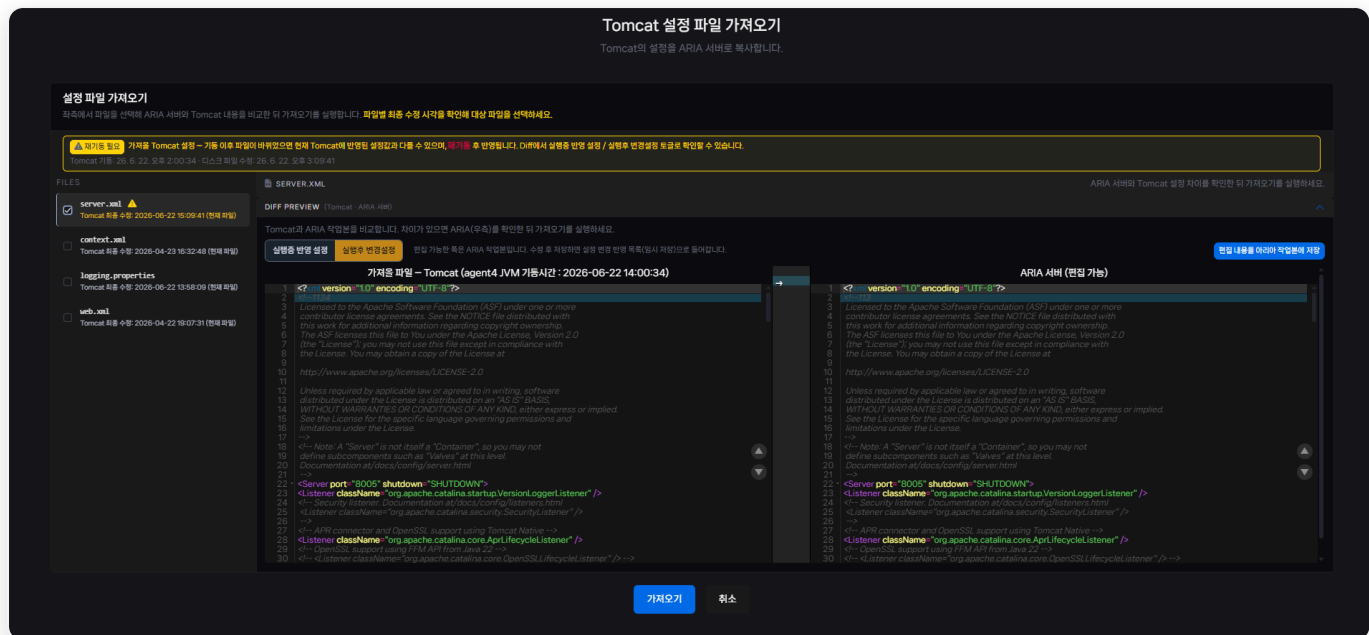


실행 중 / 실행 후 설정이 다른 경우 (Drift)

Tomcat **기동 이후** 디스크상의 설정 파일만 수동으로 바뀐 경우, **현재 JVM에 반영된 값과 디스크 파일이 다를** 수 있습니다. 이때는 상단 경고 배너와 파일 목록의 ▲ 표시가 나타납니다.

Diff 영역에서 아래 토크로 비교 기준을 바꿀 수 있습니다.

토글	의미
실행중 반영 설정	현재 기동 중인 Tomcat JVM이 사용 중인 설정 스냅샷
실행후 변경설정	기동 후 디스크에만 반영된 파일 내용



참고: 디스크만 변경된 설정은 Tomcat **재기동 전까지** 실제 서비스에 반영되지 않을 수 있습니다. 어떤 값을 ARIA에 맞출지 토글로 확인한 뒤 가져오기를 진행하세요.

5.6.5 3. 가져오기 절차

1. 에이전트 관리 화면 헤더에서 **설정 파일 가져오기**를 클릭합니다.
2. 좌측 목록에서 가져올 파일을 선택합니다. (체크박스·행 클릭)
3. Diff에서 Tomcat(좌)과 ARIA(우) 차이를 확인합니다.
4. **가져오기** 버튼을 클릭합니다.

반영 대기 설정이 있을 때

이미 ARIA에서 수정했지만 아직 **설정 변경 반영**으로 Tomcat에 보내지 않은 **반영 대기** 상태가 있으면, 가져오기 전에 확인 창이 표시됩니다.

서버에 아직 반영되지 않은 설정이 있습니다. 서버에서 파일을 가져오면 현재 반영 대기 상태가 초기화됩니다. 계속하시겠습니까?

- **서버에서 가져오기:** 반영 대기를 포기하고 Tomcat 원본으로 ARIA 작업본을 덮어씁니다.
- **취소:** 가져오기를 중단하고 기존 ARIA 작업본을 유지합니다.

완료 후

가져오기에 성공하면 **설정 파일을 정상적으로 가져왔습니다**. 메시지가 표시되고, 확인 시 화면이 새로고침됩니다.

- ARIA 작업본이 Tomcat 기준으로 갱신됩니다.
- 해당 파일을 편집 중이던 화면의 **원격 변경** 배너가 사라지고, 감지 값이 동기화됩니다.
- Tomcat 실서버 파일 자체는 **변경되지 않습니다**.

5.6.6 4. 설정 변경 반영과의 차이

기능	방향	Tomcat 즉시 반영
설정 파일 가져오기	Tomcat → ARIA	아니오 (작업본만 갱신)
설정 변경 반영	ARIA → Tomcat	예 (반영 후 재기동 포함)

설정 이력 관리에서 특정 시점으로 복원하는 것도 ARIA 작업본만 바꾸므로, 실서버 반영은 **설정 변경 반영**으로 이어야 합니다.

5.6.7 💡 활용 팁

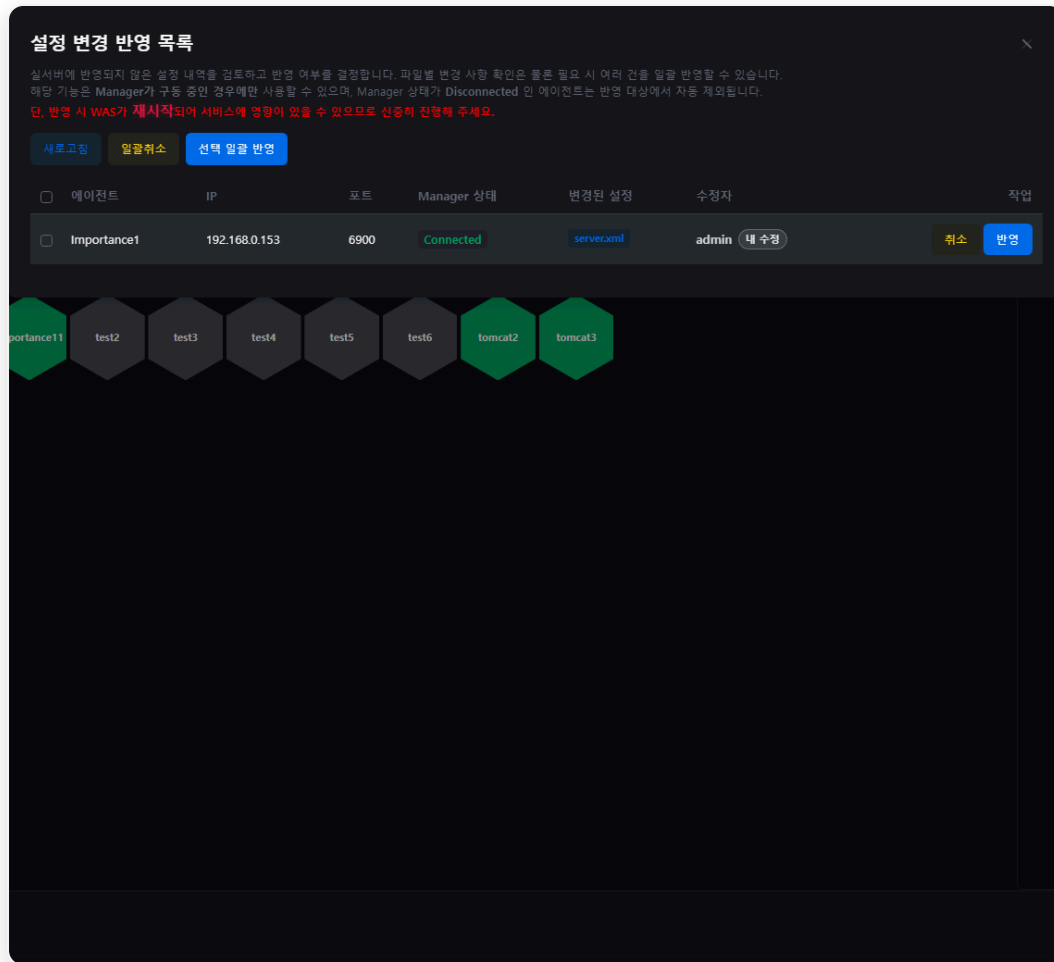
- **신규 서버 등록 직후:** Tomcat에 이미 운영 중인 설정이 있다면, 편집 전에 한 번 **가져오기**로 ARIA와 실서버를 맞춰 두세요.
- **서버 직접 수정 후:** 운영자가 SSH 등으로 `conf/` 를 수정했다면, ARIA 화면과 어긋나기 전에 가져오기로 작업본을 동기화하세요.
- **Drift ▲ 표시:** 기동 후 파일만 바뀐 경우가 의심되면 Diff 토크로 **실행중 반영 설정**과 **실행후 변경설정**을 모두 확인한 뒤 가져오세요.
- **다음 단계:** ARIA에서 수정한 내용을 Tomcat에 적용할 때는 **설정 변경 반영**을 사용하세요.

5.7 설정 변경 반영

경로: 공통 헤더 우측 상단의 **반영 대기** 아이콘 클릭

개별 에이전트의 관리창에서 설정 파일(`server.xml` , `web.xml` 등)을 수정하기 전에는 **설정 파일 가져오기**로 Tomcat 실서버 내용을 ARIA 작업본에 맞출 수 있습니다. 수정한 뒤에는 **설정 변경 반영**으로 변경 사항을 실서버에 일괄 동기화합니다. 설정 편집 방법은 [server.xml](#) 등 에이전트 설정 문서를 참고하세요.

5.7.1 화면



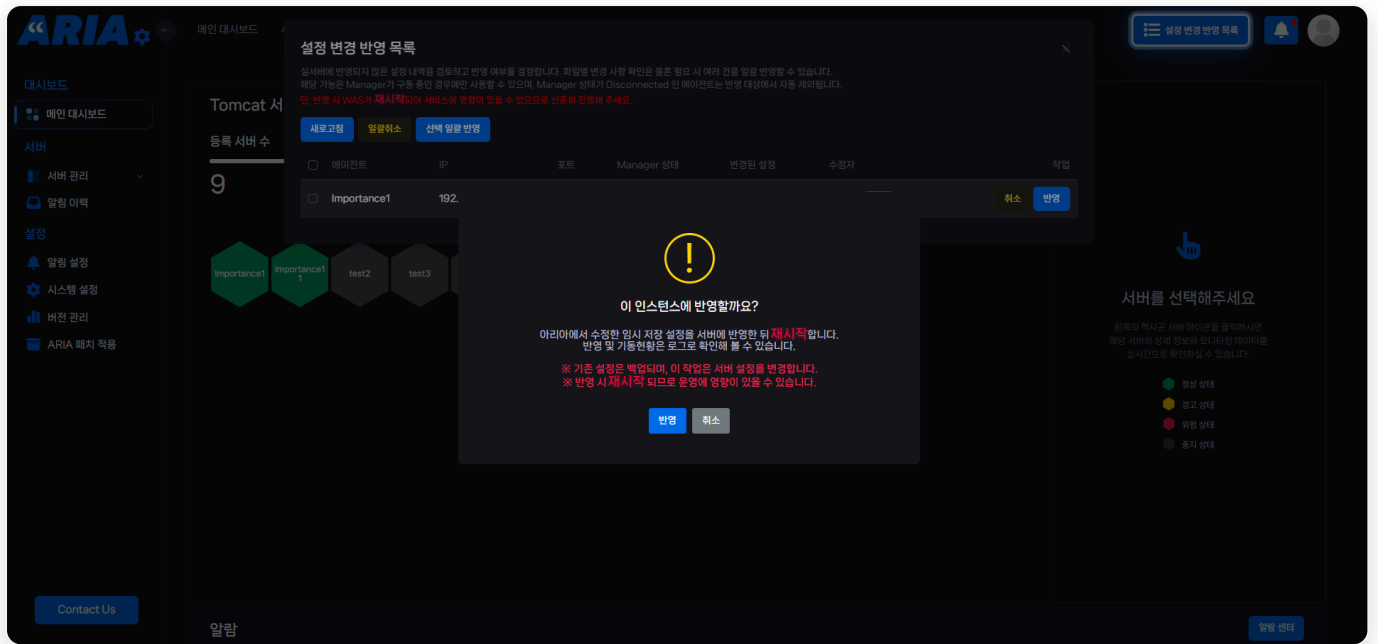
5.7.2 1. 동기화 절차

1-1. 대기 목록 확인

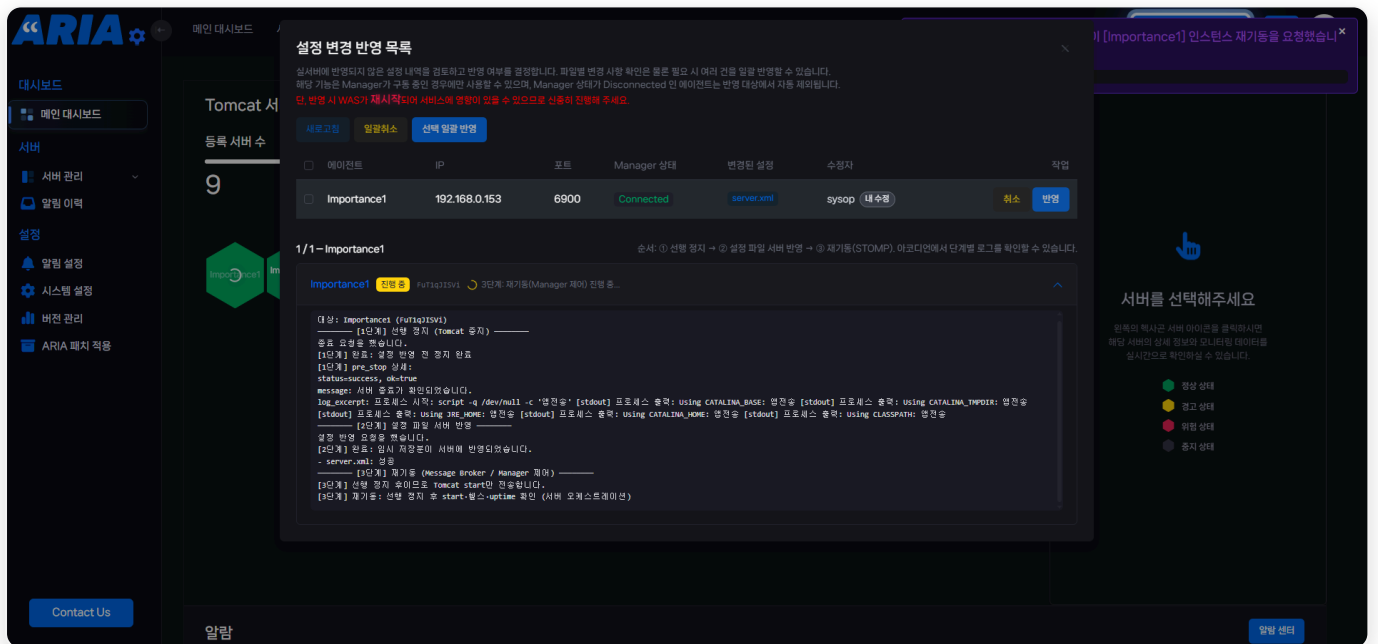
헤더 우측의 반영 대기 아이콘(숫자 배지)을 클릭하면 모달이 열립니다. 아직 실제 서버에 적용되지 않은 설정 변경 내역들이 목록에 표시됩니다. 변경된 에이전트, 파일 종류, 수정 일시를 확인하세요.

1-2. 설정 반영 실행

1. 실제 서버에 적용할 항목들의 체크박스를 선택하고 우측의 **반영** (또는 상단의 **선택 일괄 반영**) 버튼을 클릭합니다.
2. **반영 확인(Confirm)** 모달창이 나타납니다.
3. **주의:** 반영 시 WAS가 자동으로 재시작되므로 서비스에 영향이 있을 수 있음을 알리는 경고를 확인합니다.



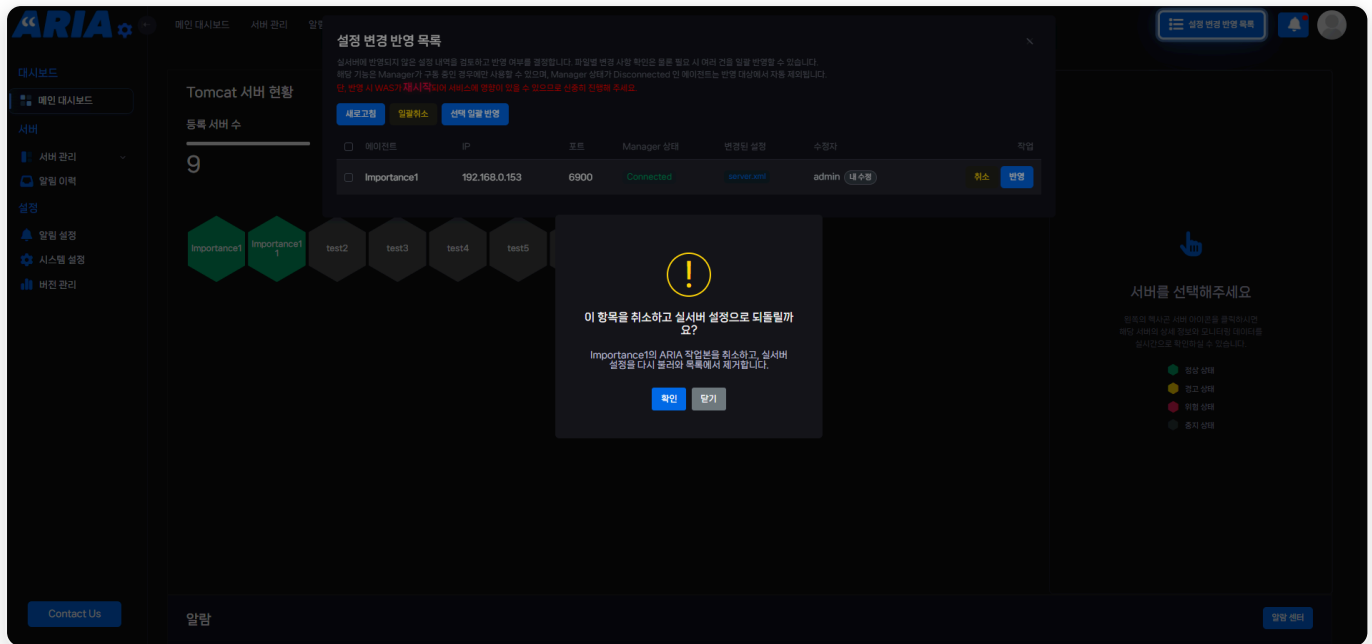
1. 반영 버튼을 클릭하면 실제 반영 작업이 시작됩니다.
2. 설정 변경 반영 목록 하단에 에이전트별 진행률과 상세 실행 로그가 실시간으로 출력됩니다.
3. 작업은 ① 선행 정지 → ② 설정 파일 반영 → ③ Manager를 통한 원격 재기동 단계로 순차 진행됩니다.



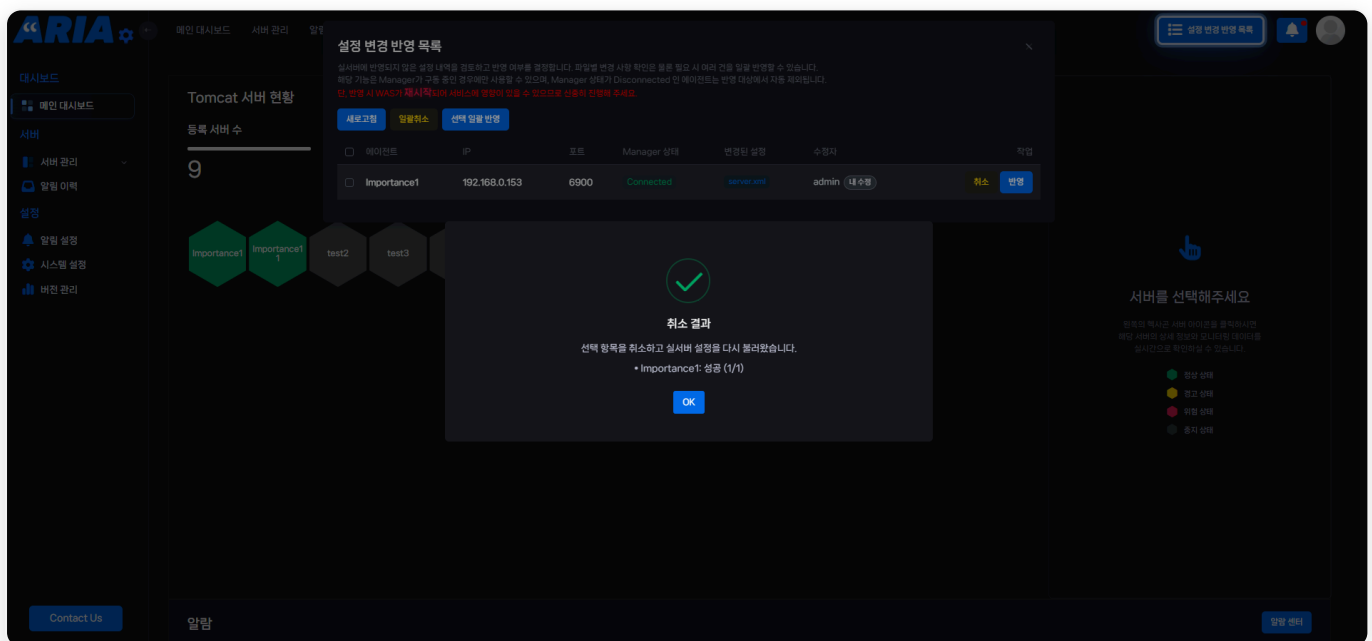
1-3. 설정 변경 취소 (원복)

작성된 임시 설정을 폐기하고, 실서버의 기존 설정을 다시 불러와 목록에서 제외할 수 있습니다.

1. 목록에서 취소하려는 서버 행 우측의 **취소** (또는 상단의 **일괄취소**) 버튼을 클릭합니다.
2. **취소 확인** 모달창이 나타납니다.



1. **확인**을 클릭하면 해당 서버의 ARIA 작업본이 취소되고 실서버 원본 설정으로 되돌아갑니다.
2. 원복 작업이 완료되면 성공 알림창이 나타나며, 완료 후 목록에서 해당 서버가 제거됩니다.



5.7.3 💡 활용 팁

- **다중 서버 설정 동기화:** 여러 서버의 설정을 개별 관리창에서 미리 변경한 뒤, 이 모달에서 한 번에 동기화하면 운영 서버에 순차적으로 빠르고 안전하게 설정을 적용할 수 있습니다.
- **자동 서비스 재기동:** 설정 반영이 완료되면 대상 Tomcat 서버를 수동으로 재시작할 필요 없이, 반영 절차상에서 **자동으로 서비스 중지 및 재기동(Start)까지 일괄 진행**됩니다. (단, 재기동 중에는 일시적인 서비스 단절이 발생하므로 운영 중인 서비스의 경우 주의가 필요합니다.)

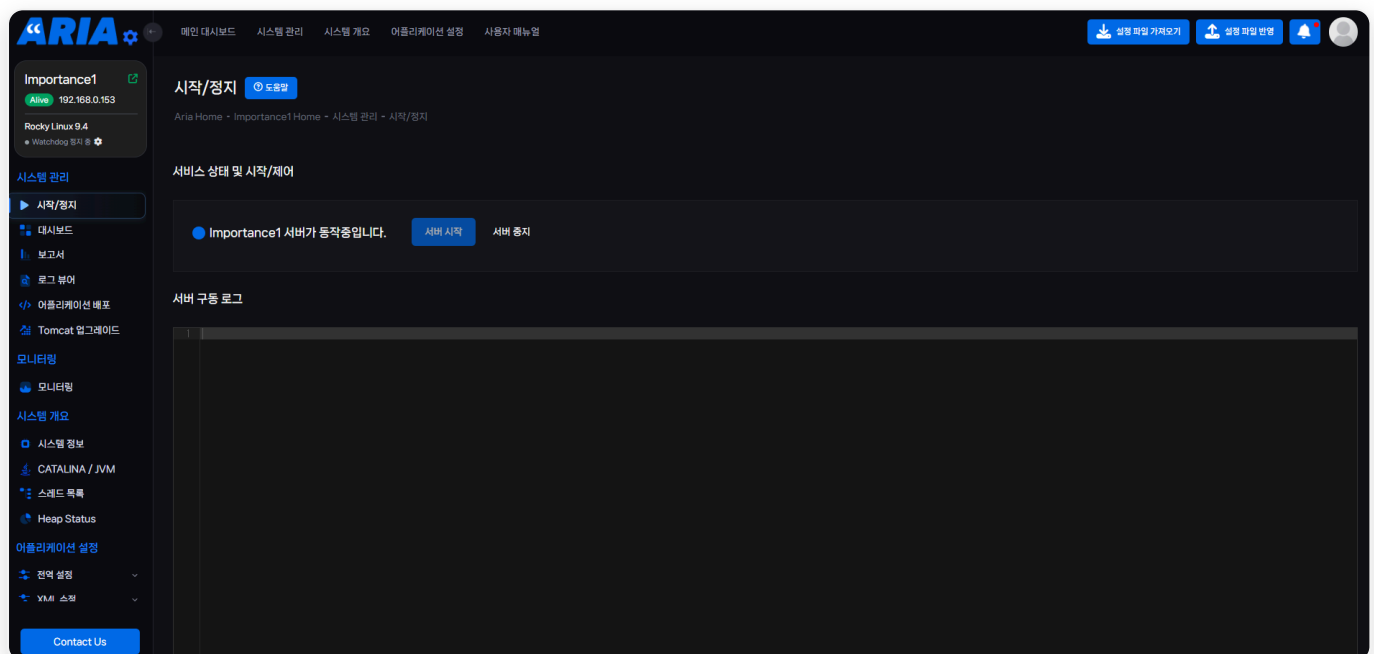
6 ⚡ 에이전트 관리

6.1 서버 상태

경로: `/servers/management/status?agent_id={agent_id}`

Tomcat 서버의 현재 운영 상태, 커넥터 통계, JVM 요약 정보를 한눈에 파악하는 페이지입니다. Tomcat Manager의 `/status` 페이지와 동일한 정보를 ARIA 웹 콘솔에서 편리하게 조회합니다.

6.1.1 화면



6.1.2 1. 서버 기본 정보

항목	설명
Tomcat 버전	현재 구동 중인 Tomcat의 버전 정보
JVM 버전	Java 런타임 버전
OS 정보	운영체제 및 아키텍처
서버 구동 시간	Tomcat 마지막 시작 이후 경과 시간

6.1.3 2. JVM 메모리 요약

항목	설명
힙 메모리 사용량	현재 사용 중인 힙 / 최대 힙 (MB)
비힙 메모리	Metaspace 등 비힙 영역 사용량

6.1.4 3. 커넥터 통계

Tomcat의 각 커넥터(HTTP, HTTPS 등)별 처리 현황을 확인합니다.

항목	설명
요청 수	커넥터가 처리한 총 요청 수
에러 수	처리 중 발생한 에러 수
처리 시간	요청 처리에 걸린 누적 시간
스레드 수	현재 사용 중인 / 최대 스레드 수

6.1.5 4. 서버 기동 및 정지 제어

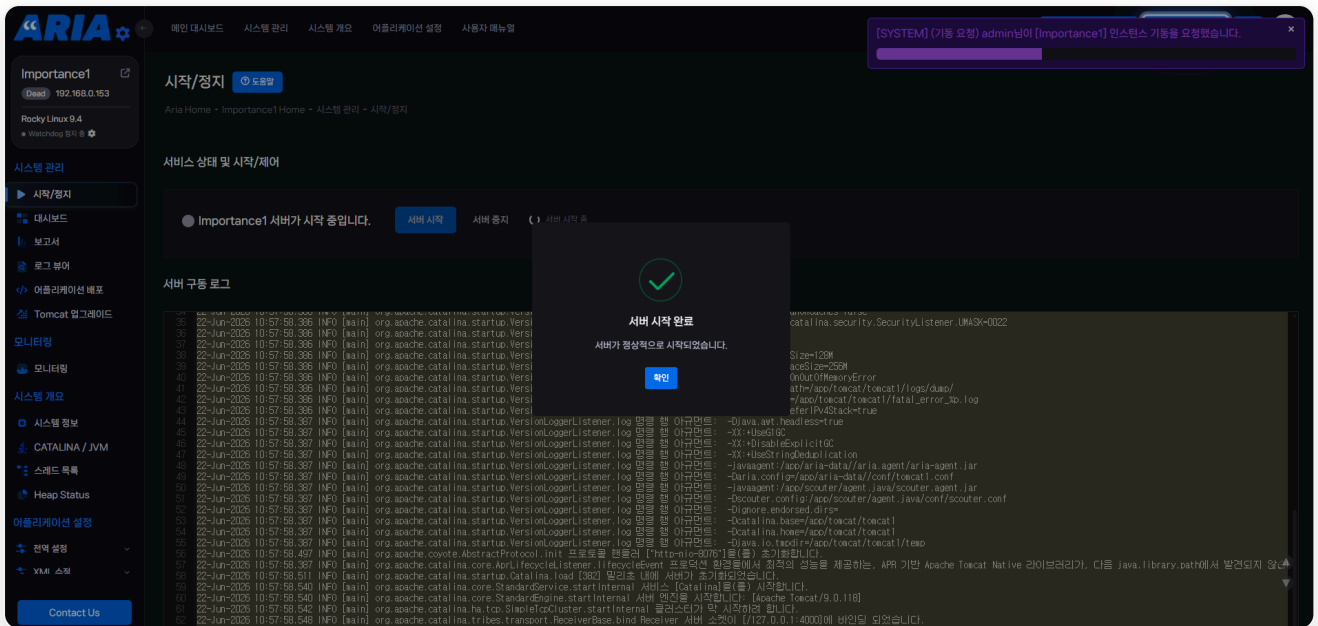
서버 상태 페이지에서는 모니터링뿐만 아니라 Tomcat 인스턴스의 기동(시작) 및 정지(중지)를 제어할 수 있는 콘솔 인터페이스를 제공합니다.

4-1. 서버 기동 (시작)

Tomcat 서버가 정지(Dead) 상태일 때 활성화되며, 시작 제어를 수행합니다.

1. 우측 상단의 **시작** 버튼을 클릭합니다.
2. 중앙의 **실시간 로그 콘솔**에 Tomcat의 구동 프로세스 로그가 실시간으로 출력됩니다.

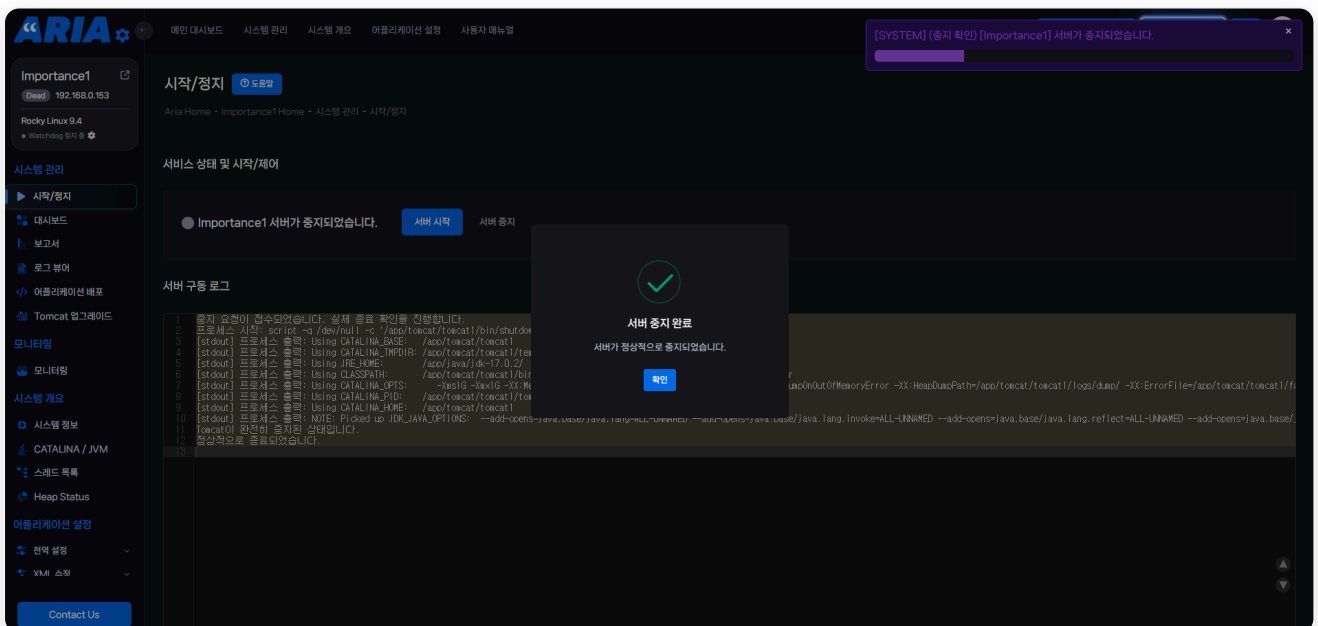
3. 구동이 완료되고 서버 상태가 기동(Alive)으로 감지되면 시작 완료 안내 모달이 표시되며 완료됩니다.



4-2. 서버 정지 (중지)

Tomcat 서버가 기동(Alive) 상태일 때 활성화되며, 중지 제어를 수행합니다.

1. 우측 상단의 **중지** 버튼을 클릭합니다.
2. 실시간 로그 콘솔에 중지 프로세스 로그가 수신됩니다.
3. **지연 시 강제 종료**: 서버 중지 처리가 30초 이상 지연될 경우, 강제 종료 의사 확인 경고 팝업이 노출됩니다. **강제 종료**를 클릭 하면 즉시 **강제 정지** 명령을 기동합니다.
4. Tomcat 인스턴스가 완전히 중지되고 서버 상태가 중지(Dead)로 감지되면 종료 완료 처리됩니다.

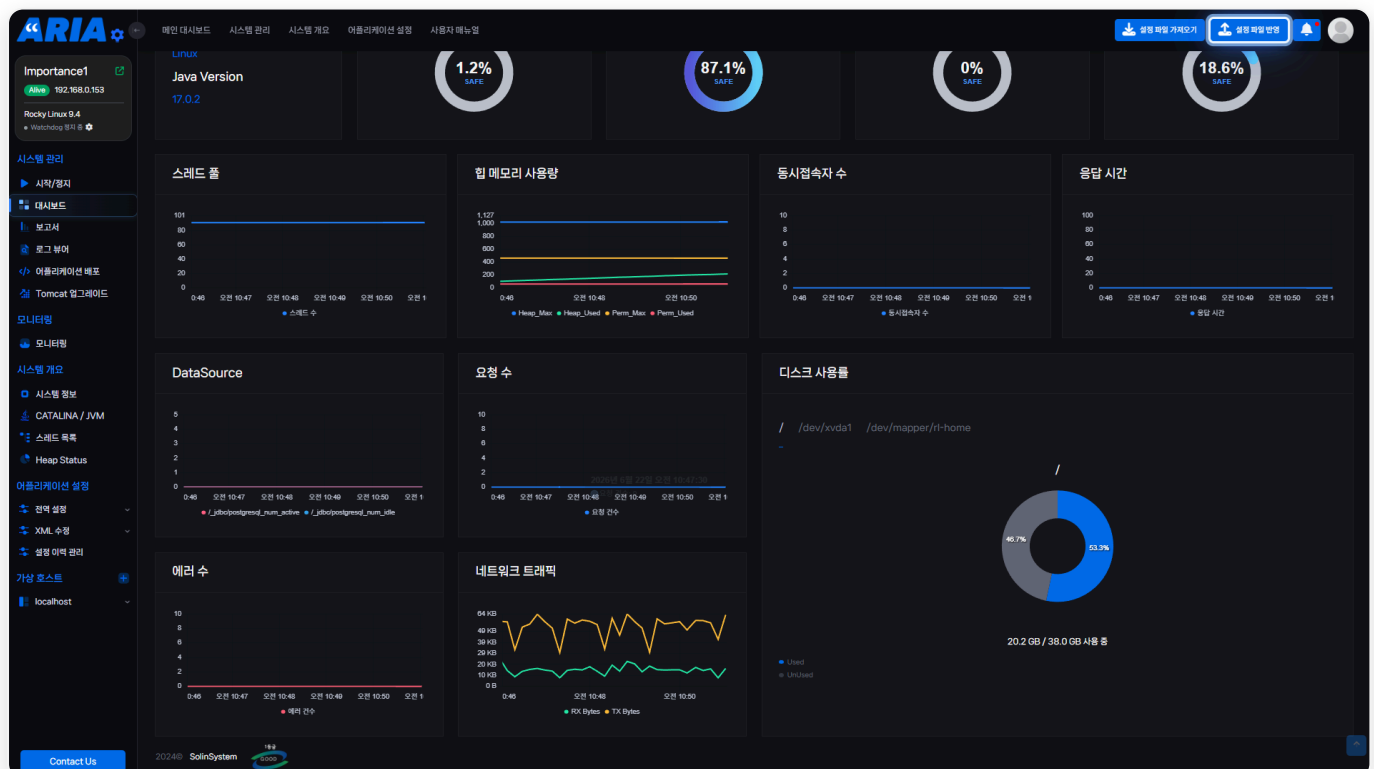
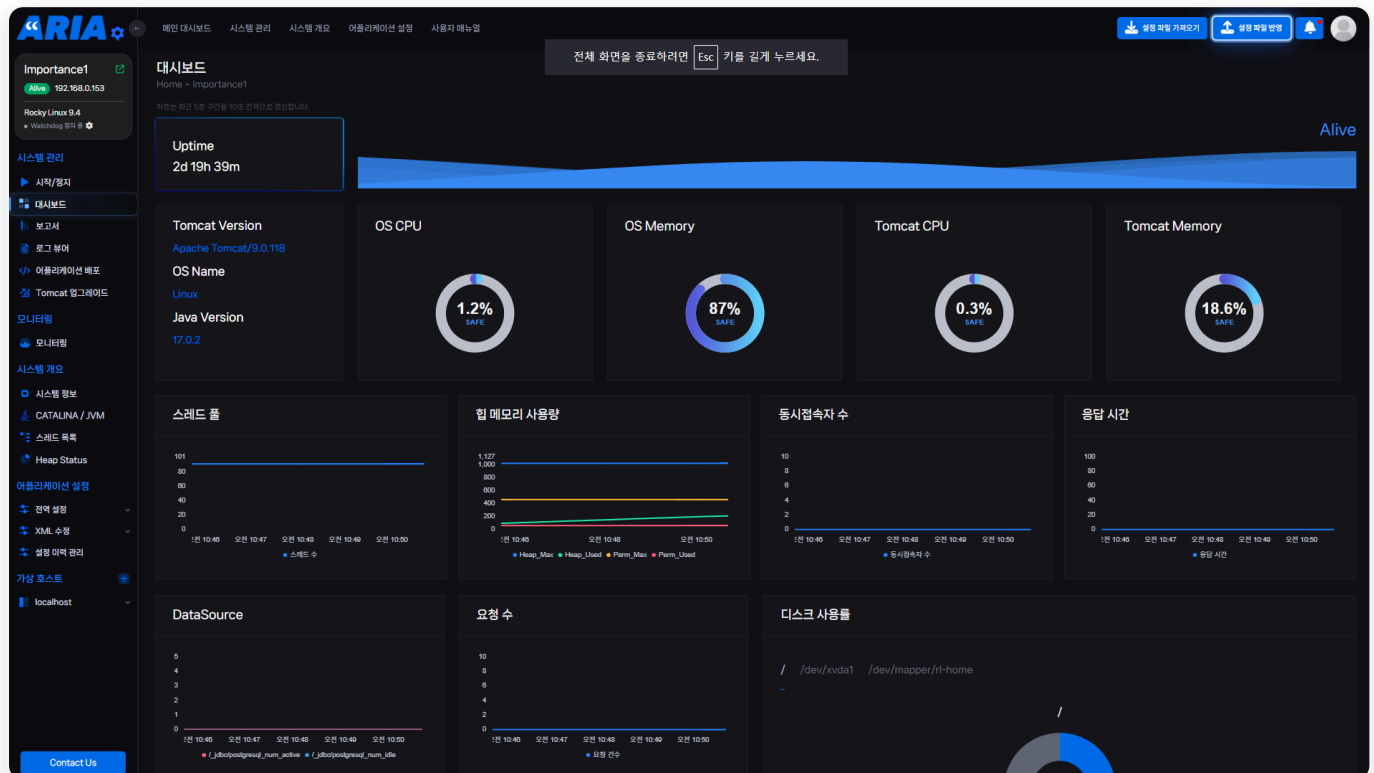


6.2 에이전트 대시보드

경로: /servers/management/dashboard?agent_id={agent_id}

선택한 단일 서버(에이전트)의 실시간 리소스 상태와 트래픽 현황을 직관적인 차트로 확인하는 종합 모니터링 페이지입니다. CPU, 메모리, 디스크 사용량 등 하드웨어 지표와 Tomcat 스레드 상태, JVM 힙 메모리 등 애플리케이션 지표를 한눈에 파악하여 장애 징후를 조기에 진단할 수 있습니다.

6.2.1 화면



6.2.2 1. 화면 구성

영역	위치	설명
서버 정보 헤더	상단	현재 서버의 OS, IP, 에이전트 버전 정보
시간 범위 선택	우상단	차트에 표시할 데이터 기간 설정
위젯 그리드	중앙	각 지표별 시계열 차트

2. 시간 범위 설정

우상단 드롭다운에서 차트에 표시할 기간을 선택합니다.

옵션	설명
5분	최근 5분 (기본값)
30분	최근 30분
1시간	최근 1시간
6시간	최근 6시간
1일	최근 24시간

6.2.3 3. 모니터링 지표

3-1. CPU 사용률

서버 전체 CPU 사용률(%)을 시계열 라인 차트로 표시합니다. 급격한 스파이크가 발생하면 특정 프로세스나 요청 급증을 확인하세요.

3-2. 메모리 사용량

물리 메모리 전체 대비 사용 중인 메모리 비율과 절대값(GB)을 표시합니다.

3-3. 디스크 I/O

디스크 볼륨별 읽기/쓰기 처리량(B/s)과 IOPS를 표시합니다. 각 파티션을 선택하여 해당 볼륨의 상세 지표만 필터링할 수 있습니다.

파티션	용도
/dev/xvda1	루트 파티션 - OS 및 기본 파일
/dev/mapper/rl-home	홈 파티션 - 사용자 데이터 및 로그

3-4. 네트워크 트래픽

초당 네트워크 수신(Rx) 및 송신(Tx) 바이트를 표시합니다.

3-5. JVM 힙 메모리

Tomcat이 사용하는 JVM의 힙 메모리 현황을 표시합니다. 힙 사용량이 지속적으로 증가하면 메모리 누수를 의심할 수 있습니다.

6.2.4 💡 활용 팁

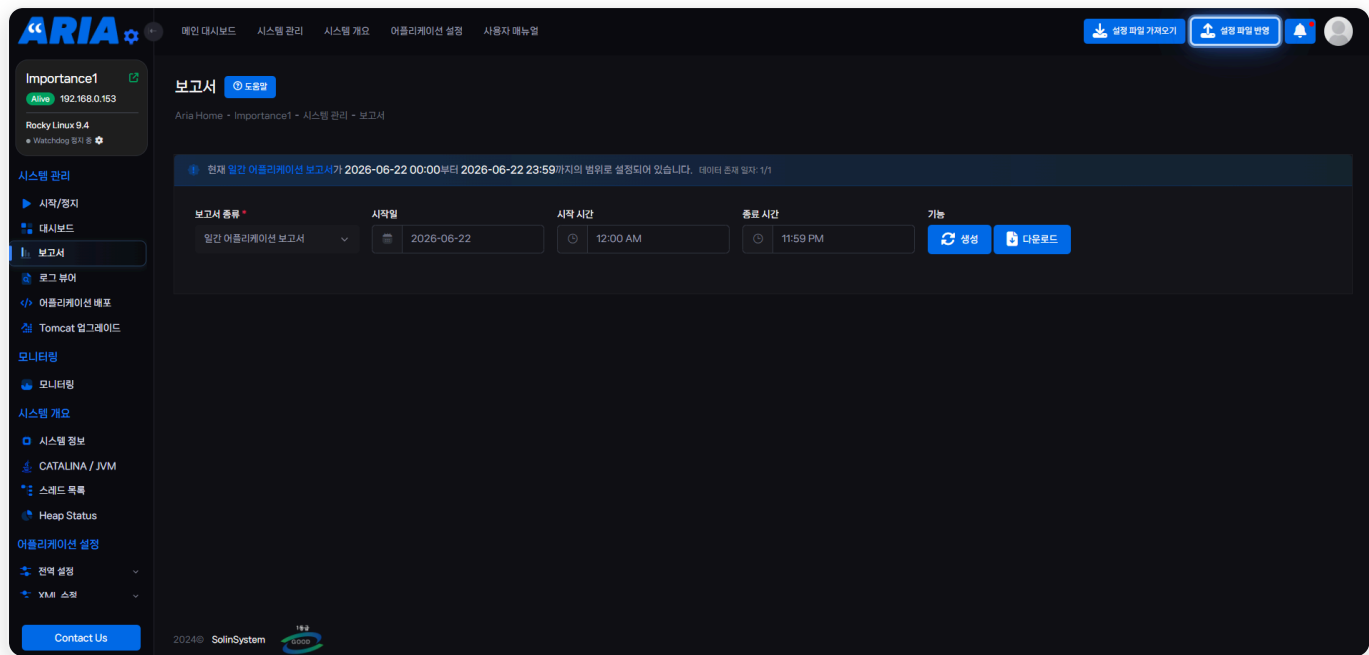
- **임계값 모니터링:** CPU 80% 이상, 메모리 90% 이상 지속 시 즉각 원인을 파악하세요.
- **디스크 풀 방지:** 홈 파티션의 디스크 사용률이 70%를 초과하면 로그 파일 정리를 검토하세요.
- **JVM 힙 누수 감지:** JVM 힙 사용량이 GC 이후에도 계속 증가하면 [힙 덤프](#)를 생성하여 분석하세요.

6.3 보고서

경로: `/servers/management/report?agent_id={agent_id}`

선택한 서버의 OS 성능 데이터 및 Tomcat 인스턴스 지표를 종합 분석하여 정기 또는 커스텀 성능 보고서를 생성하고, PDF 파일로 다운로드할 수 있는 모니터링 페이지입니다.

6.3.1 화면



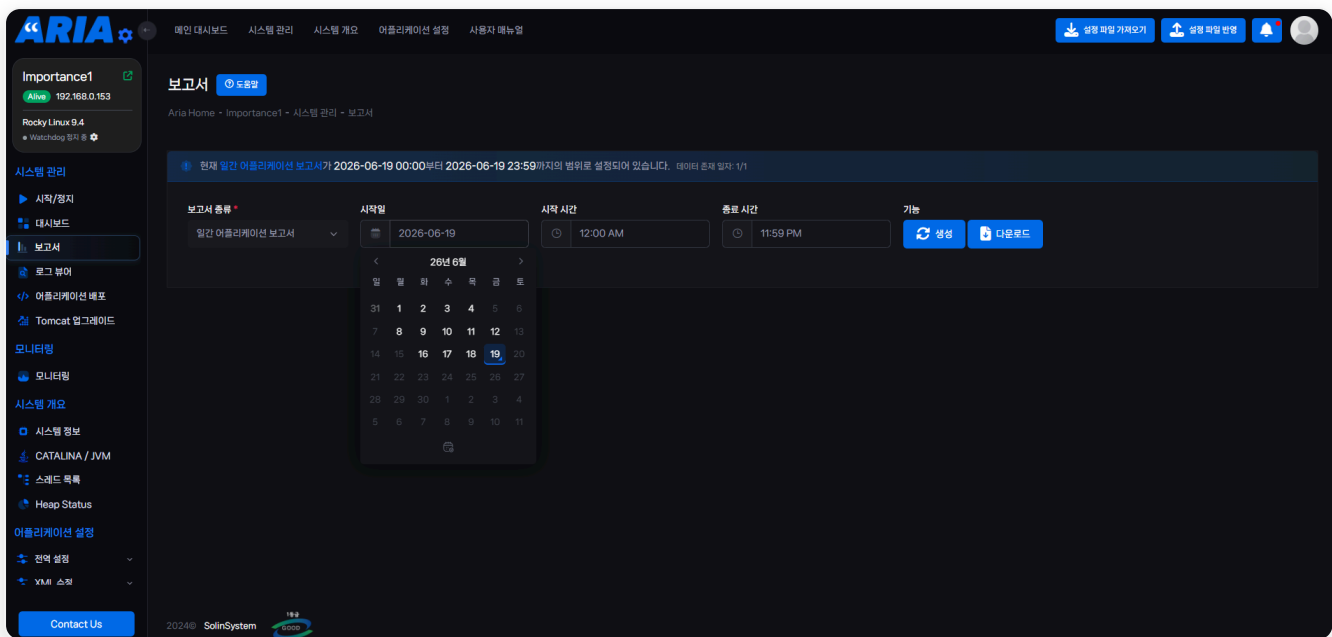
6.3.2 1. 보고서 종류 및 설정 기준

ARIA는 인프라 분석 및 경영 보고 등 용도에 맞게 4가지 유형의 보고서를 지원합니다. 보고서 종류에 따라 수집 데이터의 분석 단위 (차트 Gap)가 다르게 적용됩니다.

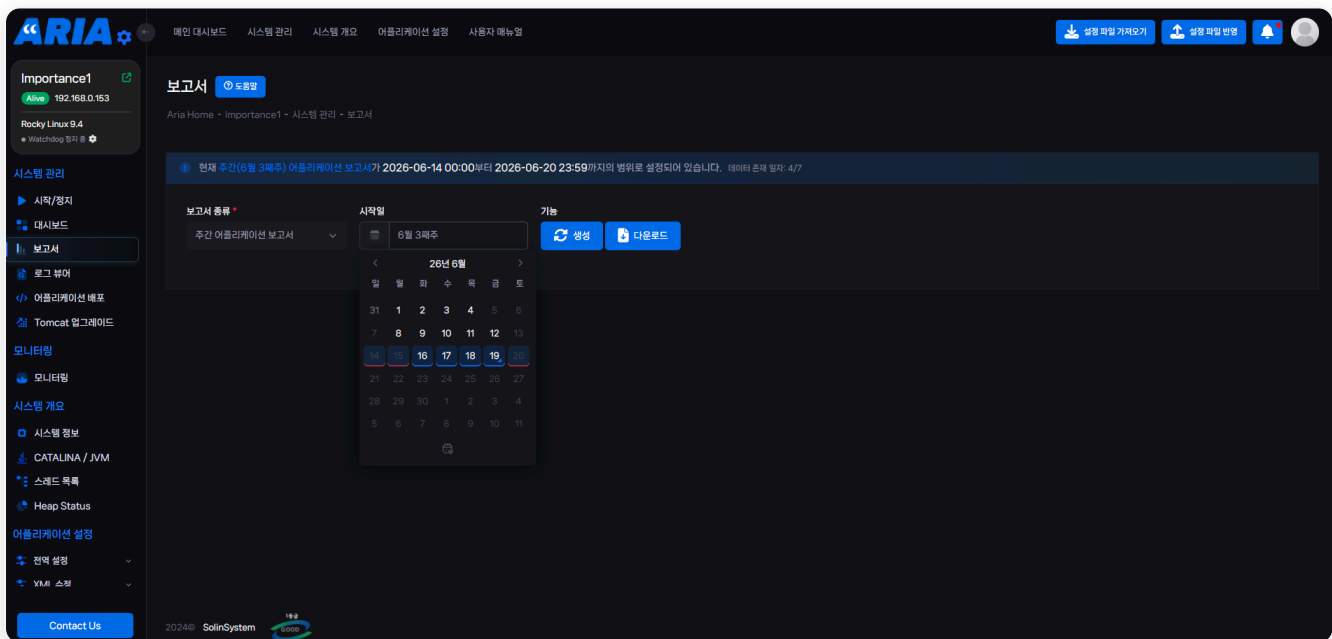
보고서 종류	설명	분석 단위 (차트 Gap)
일간 애플리케이션 보고서	특정 1일 동안의 성능 추이를 분석합니다.	5분
주간 애플리케이션 보고서	특정 1주일(7일) 동안의 성능 추이를 분석합니다.	3시간
월간 애플리케이션 보고서	특정 1개월 동안의 성능 추이를 분석합니다.	1일
사용자 정의 애플리케이션 보고서	사용자가 수동 설정한 특정 시작일~종료일 사이의 성능을 분석합니다.	5분

(각 보고서 종류별 설정 화면 예시)

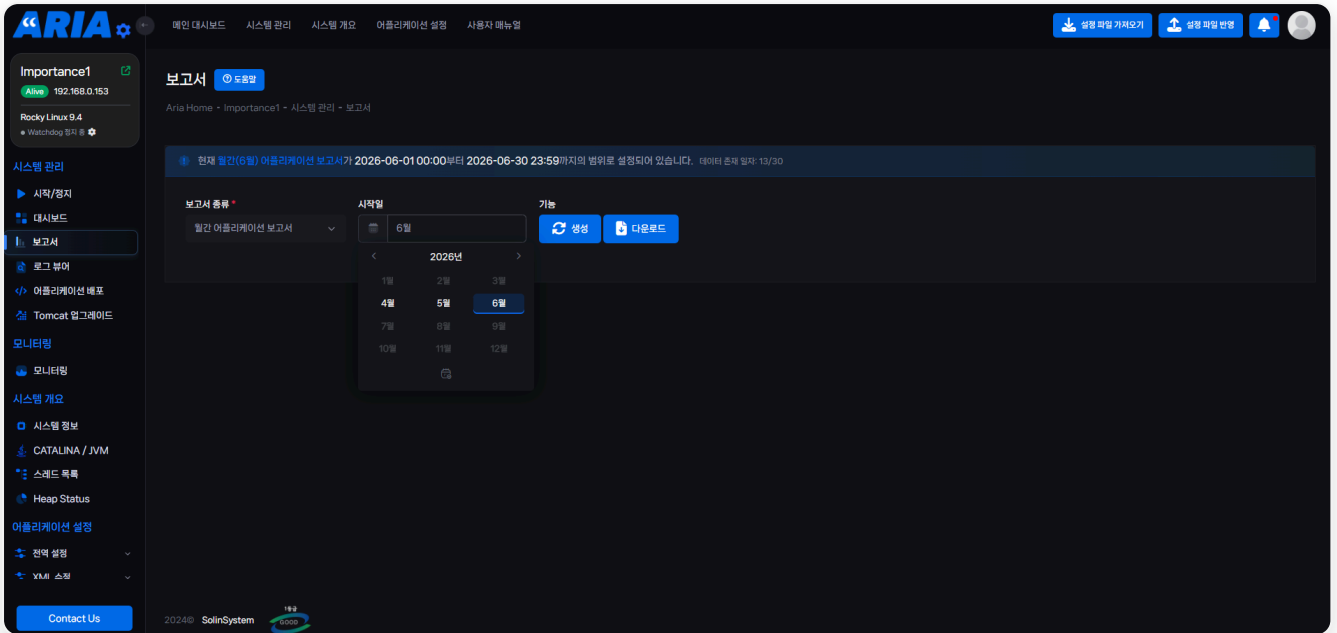
• 일간 어플리케이션 보고서 설정



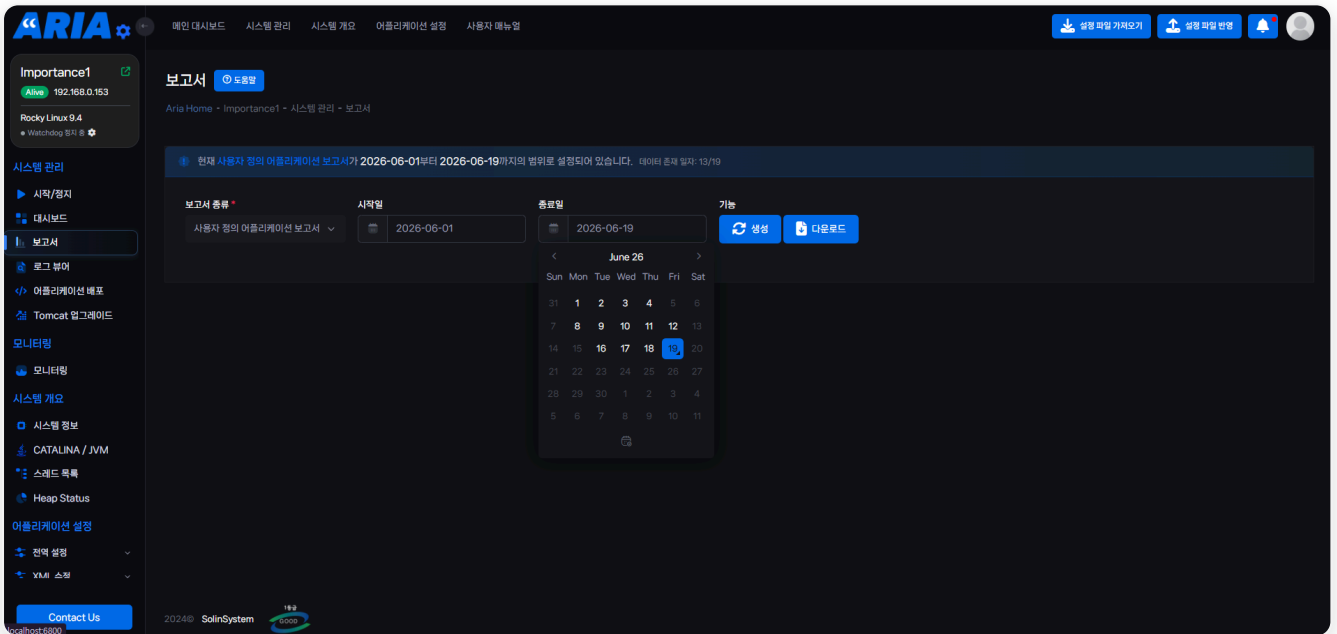
• 주간 어플리케이션 보고서 설정



- 월간 어플리케이션 보고서 설정



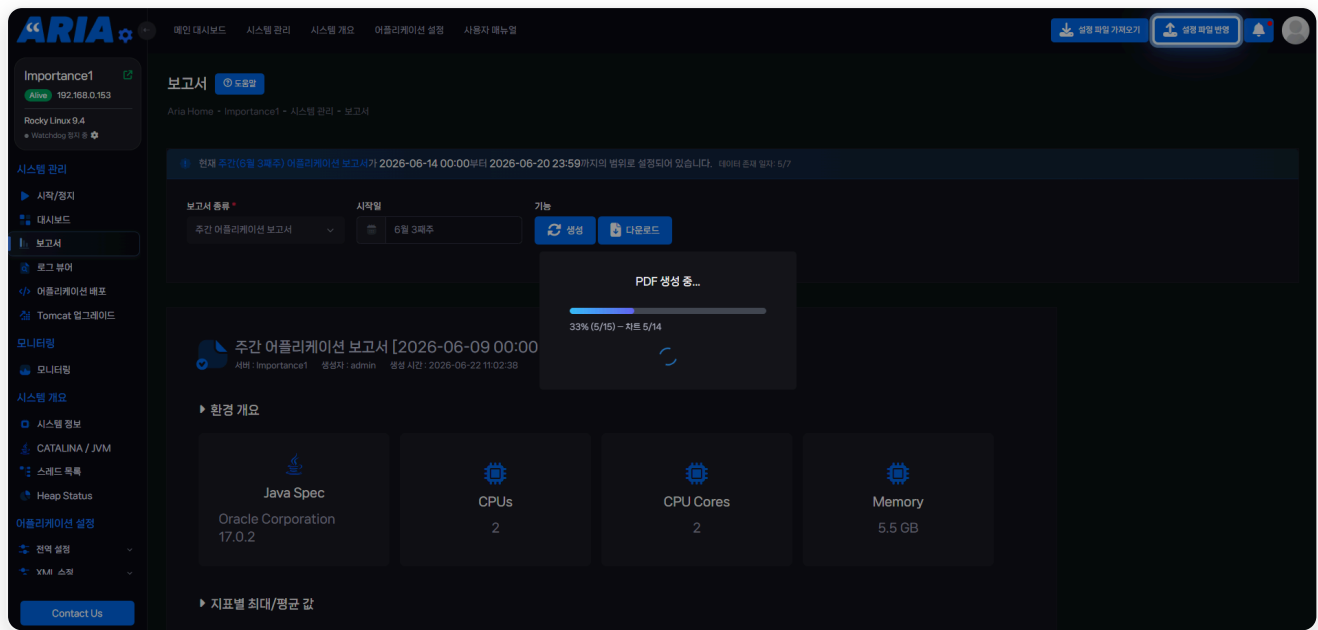
- 사용자 정의 어플리케이션 보고서 설정



2. 보고서 생성 및 다운로드 절차

1. **보고서 종류 선택:** 상단 드롭다운에서 일간/주간/월간/사용자 정의 중 하나를 선택합니다.
2. **대상 기간 및 시간 지정:** 보고서 종류에 맞춰 활성화되는 날짜 피커를 통해 대상 기간(시작일, 종료일, 시간)을 설정합니다.
3. **보고서 생성:** 우상단의 **생성** 버튼을 클릭하면 수집된 리소스 및 성능 데이터를 취합하여 실시간 보고서를 화면에 렌더링합니다.

4. **PDF 다운로드**: 보고서 생성이 완료된 후 **다운로드** 버튼을 클릭하여 출력용 PDF 문서를 로컬 기기에 저장할 수 있습니다.

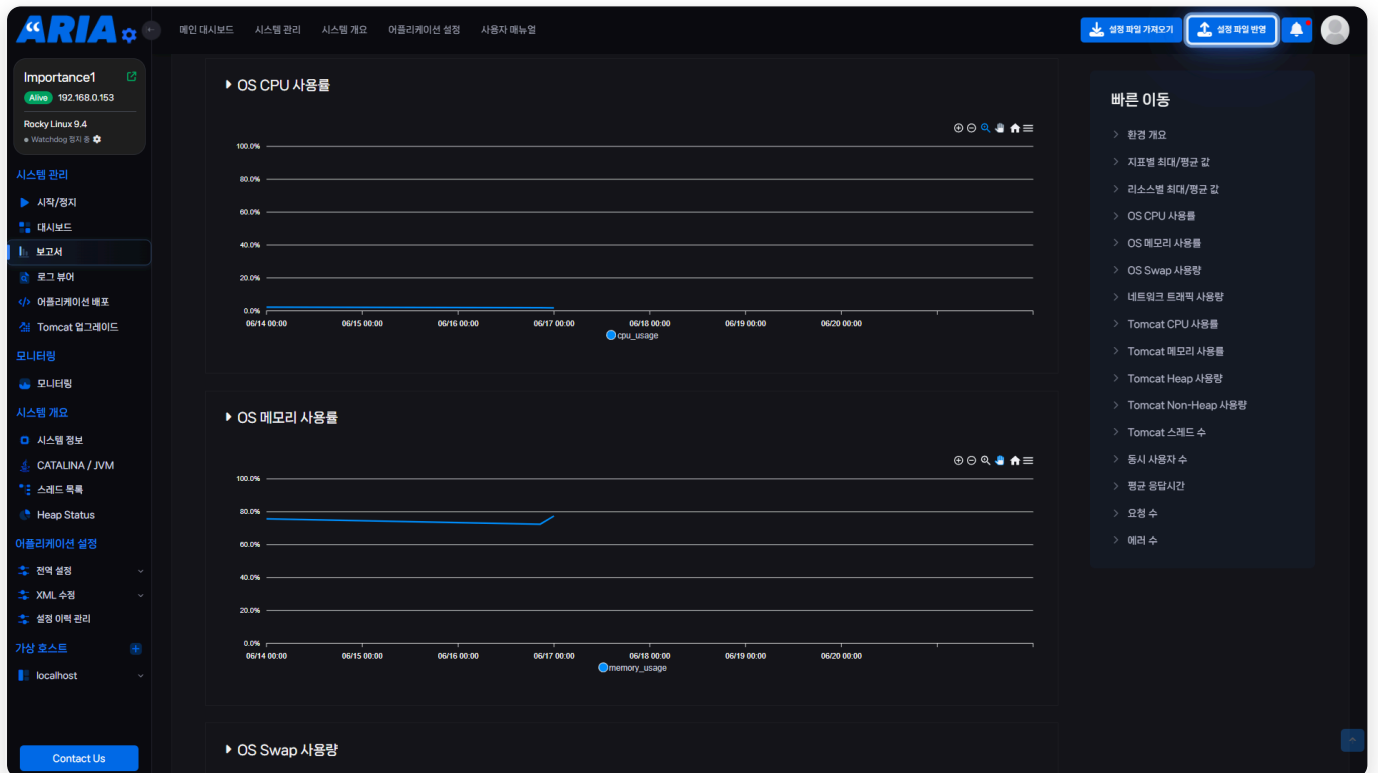
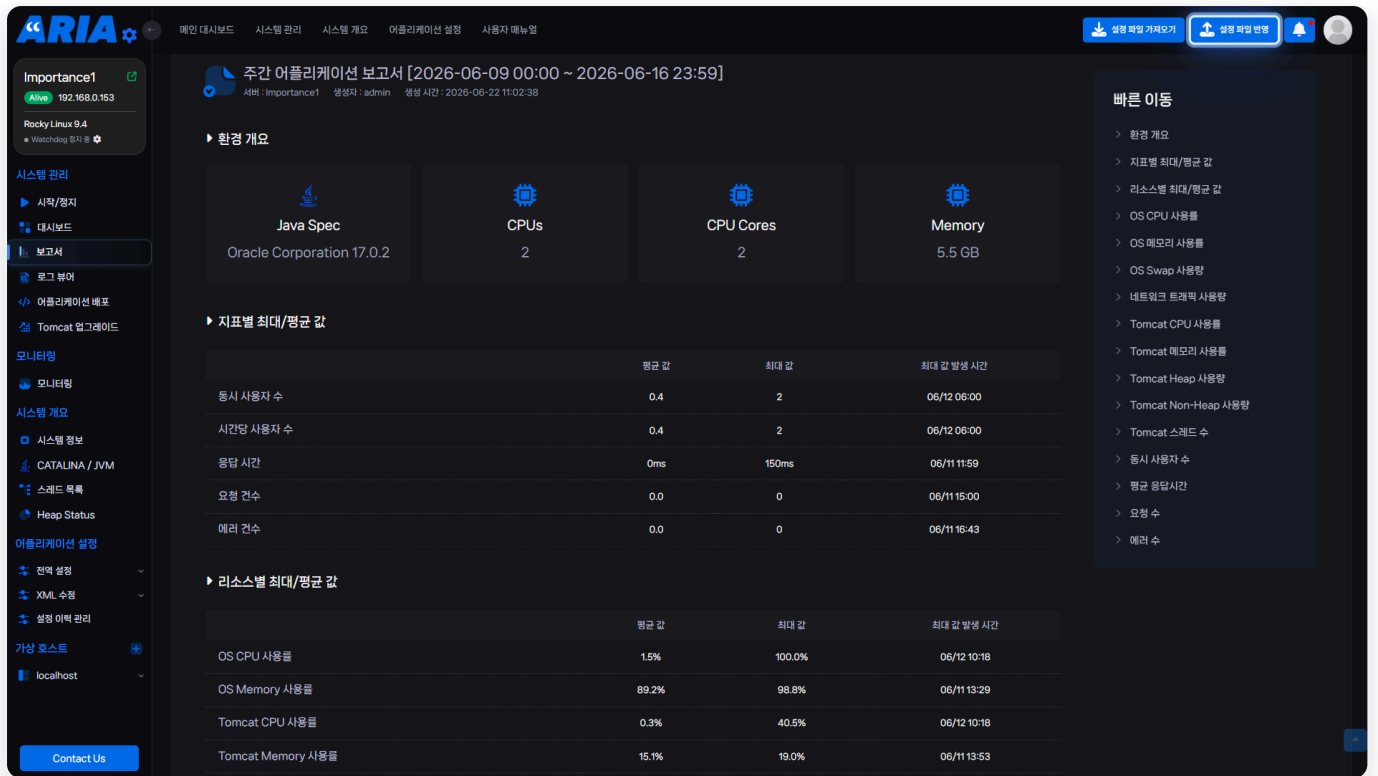


6.3.3 3. 보고서 수록 내용 및 지표

생성된 보고서에는 시스템 운영 상태를 진단할 수 있는 **요약 테이블**과 **시계열 분석 차트**가 자동으로 포함됩니다. 우측 **빠른 이동 (Quick Navigation)** 메뉴로 각 항목 위치로 바로 이동할 수 있습니다.

구분	수록 항목
요약 (①~③)	환경 개요 (Java 스펙, CPU 수·코어 수, 물리 메모리), 지표별 최대/평균 (동시·시간당 사용자, 평균 응답 시간, 요청·에러 건수), 리소스별 최대/평균 (OS/Tomcat CPU·메모리, Heap)
리소스 차트 (④~⑥)	OS CPU·메모리·Swap, 네트워크 트래픽, Tomcat CPU·메모리, Heap, Non-Heap
성능 차트 (⑦~⑮)	Tomcat 스레드 수, 동시 사용자 수, 평균 응답시간, 요청 수, 에러 수, Datasource

아래 이미지는 생성된 보고서 화면의 **예시**입니다. 스크롤 영역이 길어 한 장에 모든 항목이 들어가지 않으므로, 상·중·하 구간을 나눠 참고용으로 첨부했습니다.





6.3.4 💡 활용 팁

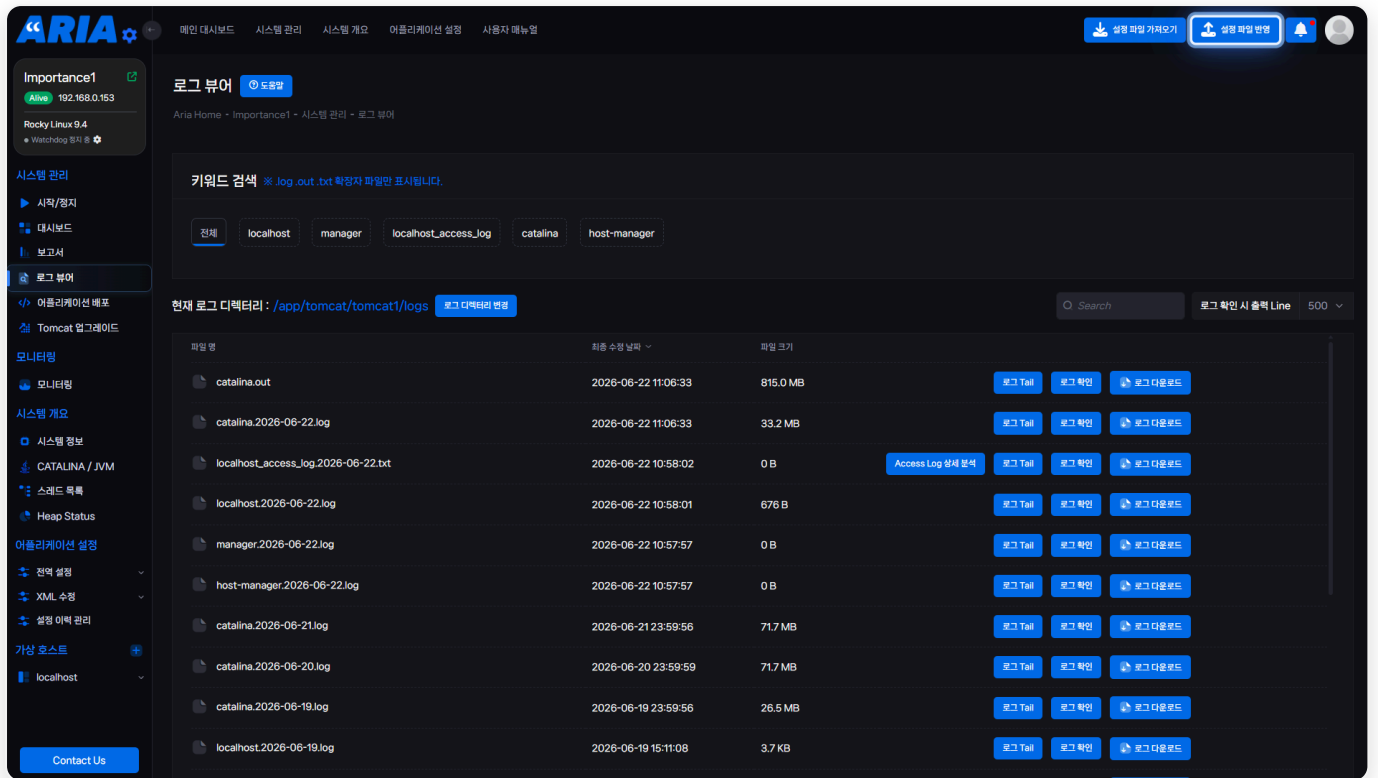
- **정기 백업 보고:** 매주 월요일 또는 매월 1일에 전주/전월 성능 보고서를 생성하여 인프라 성능 이력을 리포트 PDF 형태로 아카이빙하세요.
- **성능 임계치 분석:** 지표별 최대값 발생 시각을 감사의 알림 기록 시각과 교차 검증하면, 특정 시간대 장애의 직접적인 연관 관계를 정밀하게 파악할 수 있습니다.

6.4 로그 뷰어

경로: `/servers/management/log/viewLog?agent_id={agent_id}`

터미널(SSH)로 서버에 직접 접속하지 않고도 웹 브라우저에서 Tomcat 로그 파일(`catalina.out` , `localhost_access_log` 등)을 실시간으로 조회하고 검색하는 핵심 모니터링 페이지입니다. 에러 로그 실시간 추적(Tail), 과거 로그 다운로드, Access Log 시각 분석 도구를 제공하여 신속한 디버깅과 트래픽 분석을 지원합니다.

6.4.1 화면



6.4.2.1. 로그 카테고리 선택

상단 탭을 클릭하여 조회할 로그 파일 유형을 전환합니다.

탭	로그 파일	내용
전체	전체 로그	모든 카테고리 통합 조회
catalina	catalina.out	Tomcat 구동 및 애플리케이션 표준 출력/에러
localhost	localhost.log	localhost 컨텍스트 전용 로그
localhost_access_log	localhost_access_log.*.txt	HTTP 요청 기록 (Access Log)
manager	manager.log	Tomcat 관리자 콘솔 내부 로그
host-manager	host-manager.log	가상 호스트 관리 내부 로그

추천: 장애 발생 시 가장 먼저 `catalina` 탭을 선택하여 `Exception` 이나 `Error` 키워드로 검색하세요.

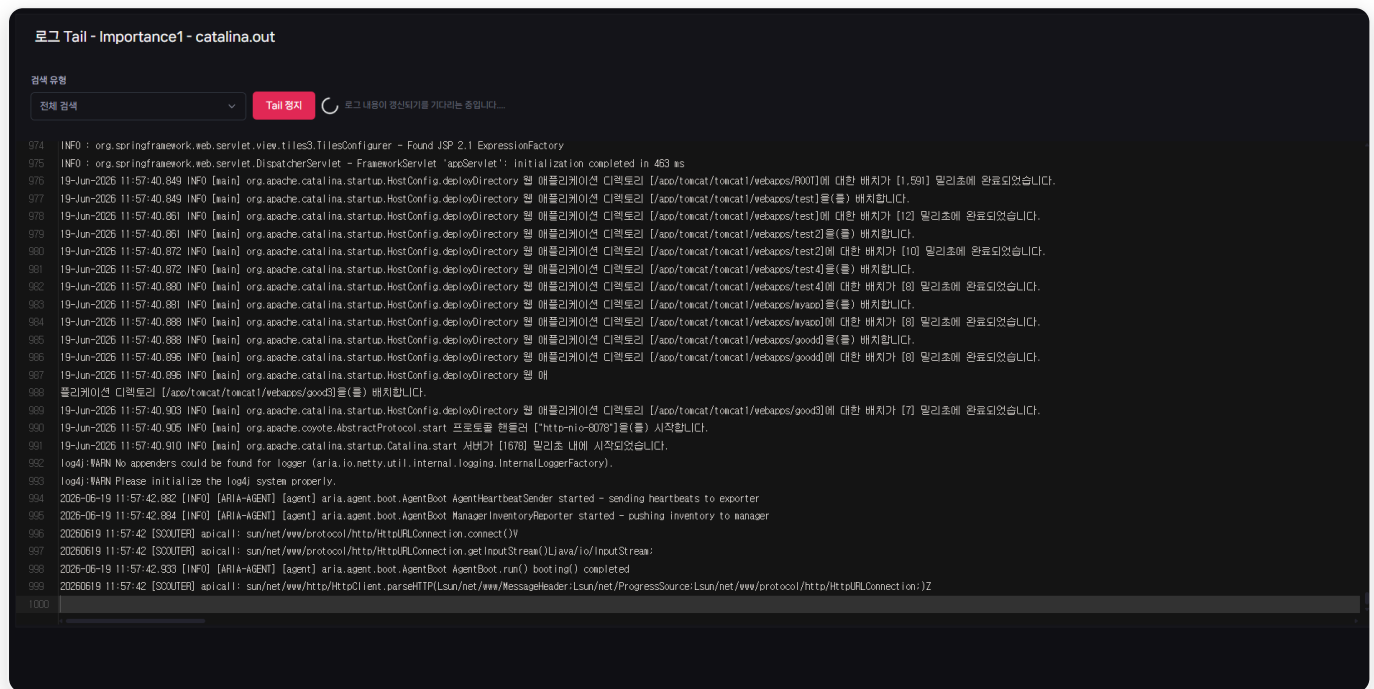
2. 로그 디렉터리 변경

Tomcat의 기본 로그 경로(`CATALINA_BASE/logs`) 외에 커스텀 경로의 로그를 조회할 때 사용합니다.

1. 로그 디렉터리 변경 버튼을 클릭합니다.
2. 조회할 로그 디렉터리 경로를 입력합니다.
3. 확인하면 해당 경로의 로그 파일 목록으로 전환됩니다.

6.4.3 3. 실시간 로그 추적 (Tail)

`tail -f` 명령어와 동일한 기능으로, 로그가 실시간으로 화면에 계속 추가됩니다.



1. 로그 Tail 버튼을 클릭하여 활성화합니다.
2. 서버에 로그가 기록될 때마다 화면 하단에 자동으로 추가됩니다.
3. 다시 클릭하면 Tail이 중지됩니다.

활용: 배포 직후나 장애 재현 테스트 시 Tail 모드를 활성화하여 에러를 실시간으로 캐치하세요.

6.4.4 4. 로그 검색 (정적 조회)

특정 키워드나 날짜 범위로 로그를 필터링하여 조회합니다.

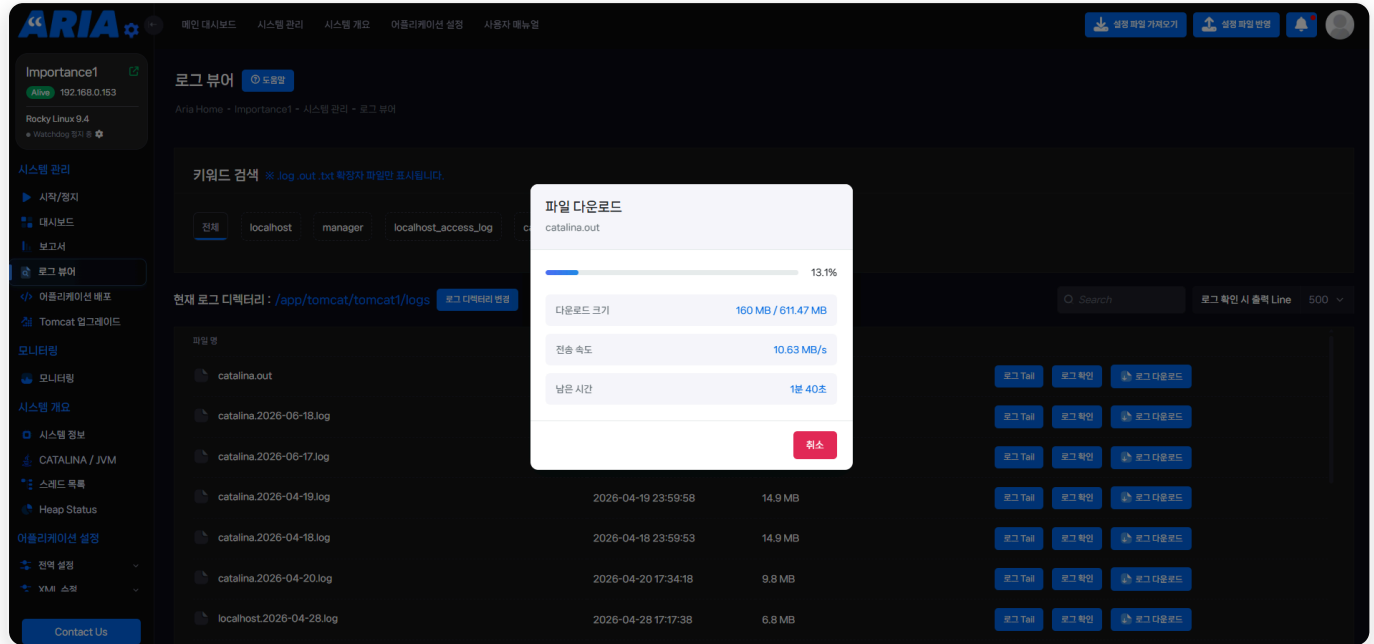
1. 상단 검색 필드에 키워드(`Exception`, `ERROR`, 특정 클래스명 등)를 입력합니다.
2. 날짜 범위를 선택합니다.
3. 로그 확인 버튼을 클릭합니다.
4. 조건에 맞는 라인만 필터링되어 표시됩니다.

6.4.5 5. 로그 다운로드

현재 선택된 로그 파일 전체를 로컬 PC로 다운로드합니다.

로그 다운로드 버튼을 클릭하면 원본 로그 파일(.txt, .log)이 즉시 다운로드됩니다.

오프라인에서 분석하거나 로컬 텍스트 에디터로 정밀 검색할 때 활용합니다.



6.4.6 6. Access Log 상세 분석

localhost_access_log 탭이 선택된 상태에서 제공되는 고급 시각화 분석 기능입니다. 텍스트 형태의 접근 로그 파일 데이터를 파싱하여 트래픽 추이, 상태 코드 분포, 병목 URI 등을 한눈에 모니터링할 수 있는 그래픽 리포트를 실시간 생성합니다.

6-1. 지원 로그 포맷(Pattern) 및 불일치 경고

Access Log 상세 분석을 정상적으로 사용하려면 Tomcat 서버의 가상 호스트(Virtual Host)에 설정된 Access Log의 출력 규격이 ARIA의 분석 포맷과 일치해야 합니다. ARIA는 아래 두 가지 로그 포맷 패턴을 지원합니다.

1) 지원 포맷 패턴 목록

1. 표준 Combined 기반 패턴 (요청 처리 속도 포함)

2. 설정 패턴: %h %l %u %t "%r" %s %b %D

3. 포맷 예시: 127.0.0.1 - - [19/Sep/2025:09:56:02 +0900] "GET /index.html HTTP/1.1" 200 1204 45

4. 필드 설명: 클라이언트 IP(%h), 신원 정보(%l), 사용자명(%u), 타임스탬프(%t), 요청 첫 행("%r"), 상태 코드(%s), 전송 바이트 수(%b), 요청 처리 시간(%D - ms/μs)

5. 확장 헤더 상세 패턴 (Referer, User-Agent 및 접두사 포함)

6. 설정 패턴: %h %l %u %t %r {%Referer%i} {%User-Agent%i} length-%b status-%s dur-%D

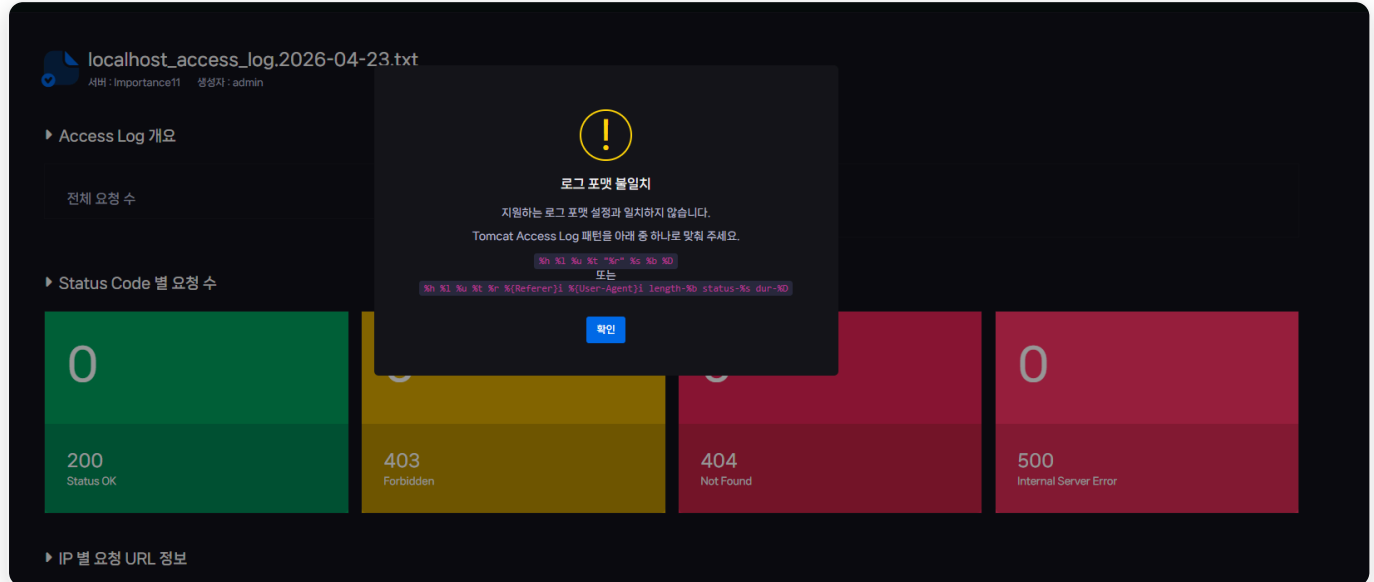
7. 포맷 예시: 127.0.0.1 - - [19/Sep/2025:09:56:02 +0900] GET /index.html HTTP/1.1 http://localhost/ Mozilla/5.0 length-1204 status-200 dur-45

8. 필드 설명: 기본 필드 외 HTTP Referer({%Referer%i})와 User-Agent({%User-Agent%i}) 헤더 정보가 기록되며, 바이트 수(length-), 상태 코드(status-), 소요시간(dur-)에 명확한 식별용 접두사가 부여된 형태입니다.

2) 로그 포맷 불일치 경고 모달

만약 Tomcat 설정 파일(`server.xml`)의 가상 호스트(`Host`) 태그 내 `AccessLogValve` 에 설정된 `pattern` 속성과 ARIA 상세 분석 모듈이 기대하는 포맷 규격이 일치하지 않을 경우, 분석 실행 시 파싱 에러를 미연에 방지하고자 아래와 같이 경고 모달이 노출됩니다.

이 오류가 발생하면 [가상 호스트 - Access Log](#)에서 `AccessLogValve` 의 `pattern` 을 아래 지원 포맷 중 하나로 변경한 뒤 [설정 변경 반영](#)으로 적용하세요.



6-2. 상세 분석 제공 화면 및 지표 정리

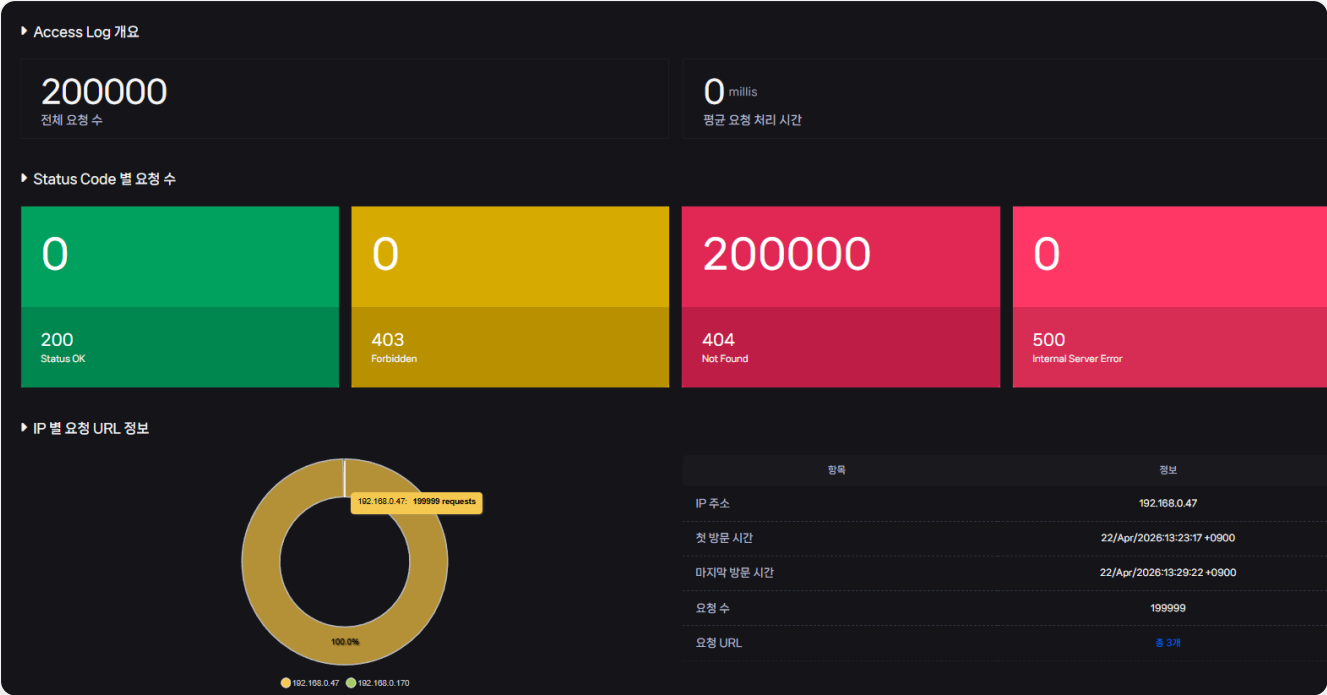
Access Log 파싱이 성공적으로 완료되면 4가지 대분류 영역으로 나누어 정밀 시각화 분석 정보를 제공합니다.

㉠ 기본 트래픽 통계 및 시간별 추이

전반적인 트래픽 규모와 주요 성능 및 에러 여부를 직관적으로 보여주며, 시간별 요청 흐름을 차트로 나타냅니다.

- **총 요청 수 (Total Requests):** 분석 대상 로그 전체에서 파싱된 총 HTTP 요청 건수
- **평균 응답 속도 (Average Response Time):** 전체 요청의 처리 소요 시간 평균값 (ms)
- **주요 응답 상태코드 집계:** HTTP `200 OK`, `403 Forbidden`, `404 Not Found`, `500 Internal Server Error` 에 해당하는 누적 응답 건수를 카드 인터페이스로 제공

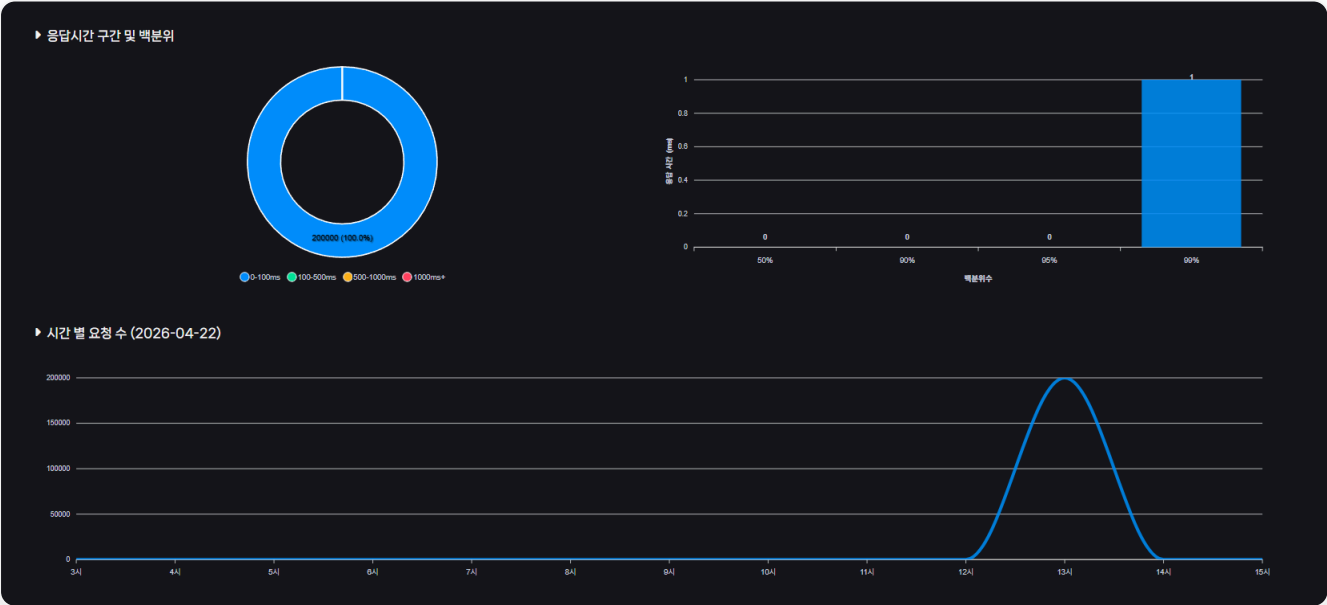
- 시간대별 요청 수 추이 차트: 24시간대별(0시~23시) 총 요청 횟수 변화 추이를 보여주는 꺾은선(Line) 그래프



② HTTP 응답 상태 코드 및 지연 시간 분포

서버가 반환한 응답 코드의 전체적인 비율과 요청 처리에 걸린 소요 시간의 구간별 지연을 다차원 시각화합니다.

- 응답 시간 분포 도넛 차트: 전체 요청에 대한 처리 지연 속도를 0-100ms (빠름), 100-500ms (보통), 500-1000ms (느림), 1000ms+ (매우 느림)의 4개 구간으로 분류한 백분율 도넛 차트
- 응답 시간 백분위수(Percentile) 차트: 50% (Median), 90%, 95%, 99% 사용자의 실제 응답 지연 시간(ms) 분포를 나타내는 막대(Bar) 차트로, 극단적인 응답 지연(Tail Latency) 현황 추적



③ 요청 클라이언트 IP 및 HTTP 메소드 점유율

서비스에 가장 많은 트래픽을 유발하는 원격 접속 정보와 사용 빈도가 높은 HTTP 메소드(GET, POST, PUT, DELETE 등) 점유율을 추적합니다.

- **클라이언트 IP 별 요청 도넛 차트:** 서버에 접근한 개별 클라이언트 IP들의 요청 횟수 비중 시각화
- **IP 상세 정보 표시:** 도넛 차트의 특정 IP 영역을 클릭하면 해당 IP의 첫 방문 시각, 마지막 방문 시각, 총 요청 수, 고유 방문 URI 개수 등의 세부 이력이 연동 테이블로 출력됨
- **방문 목록 확인:** 특정 IP가 호출한 모든 URI 경로 리스트를 팝업 모달창을 통해 탐색 가능

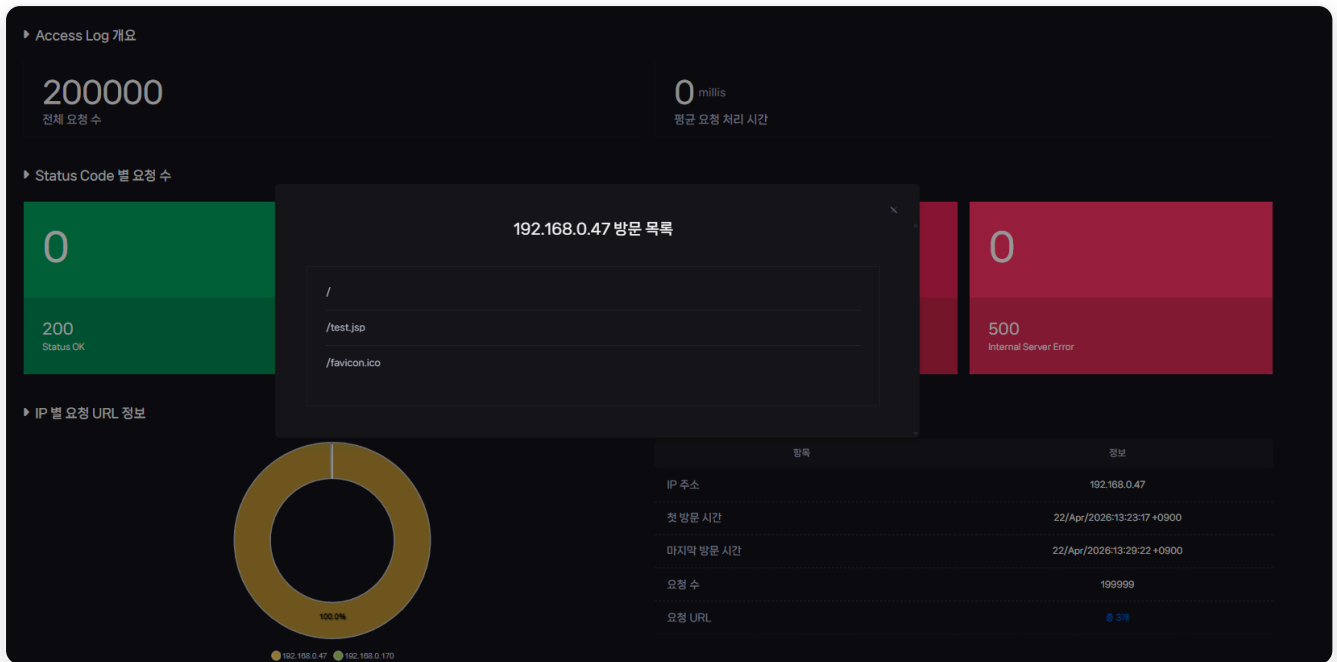
▶ TOP 10 요청 수 URL	
URL	요청 수
/	199997
/test.jsp	2
/favicon.ico	1
▶ TOP 10 평균 응답시간 URL	
URL	평균 응답 시간
/test.jsp	1 ms
/	0 ms
/favicon.ico	0 ms
▶ SQL Injection 의심 URL	
URL	기록 없음
▶ WordPress 공격 의심 URL	
URL	기록 없음

④ URI(Request URL) 성능 및 보안 위협 분석

가장 빈번하게 호출되는 엔드포인트와 응답 지연이 큰 병목 구간, 그리고 외부 공격 시도로 의심되는 비정상 접근 리스트를 분석합니다.

- **가장 많이 호출된 상위 10개 URI:** 호출 빈도가 가장 높은 URI 10개 목록 및 호출 횟수 (총 트래픽 분석에 유용)
- **평균 응답 속도가 느린 상위 10개 URI:** 평균 처리 시간(ms)이 가장 긴 URI 10개 목록 및 응답 시간 (성능 튜닝 대상 API 식별)
- **보안 위협 탐지 목록:** 로그 데이터의 URI 경로 내 특정 악성 패턴을 실시간 매칭하여 탐지
- **SQL Injection 시도:** `union`, `select`, `'` 등 데이터베이스 탈취 구문이 포함된 URI
- **Wordpress 취약점 스캔:** `wp-`, `wordpress` 등 워드프레스 플랫폼 표적 스캔 접근 URI

- **Cross-Site Scripting (XSS) 시도:** `<script`, `javascript:` 등 브라우저 악성 스크립트 실행 패턴 포함 URI



6.4.7 💡 활용 팁

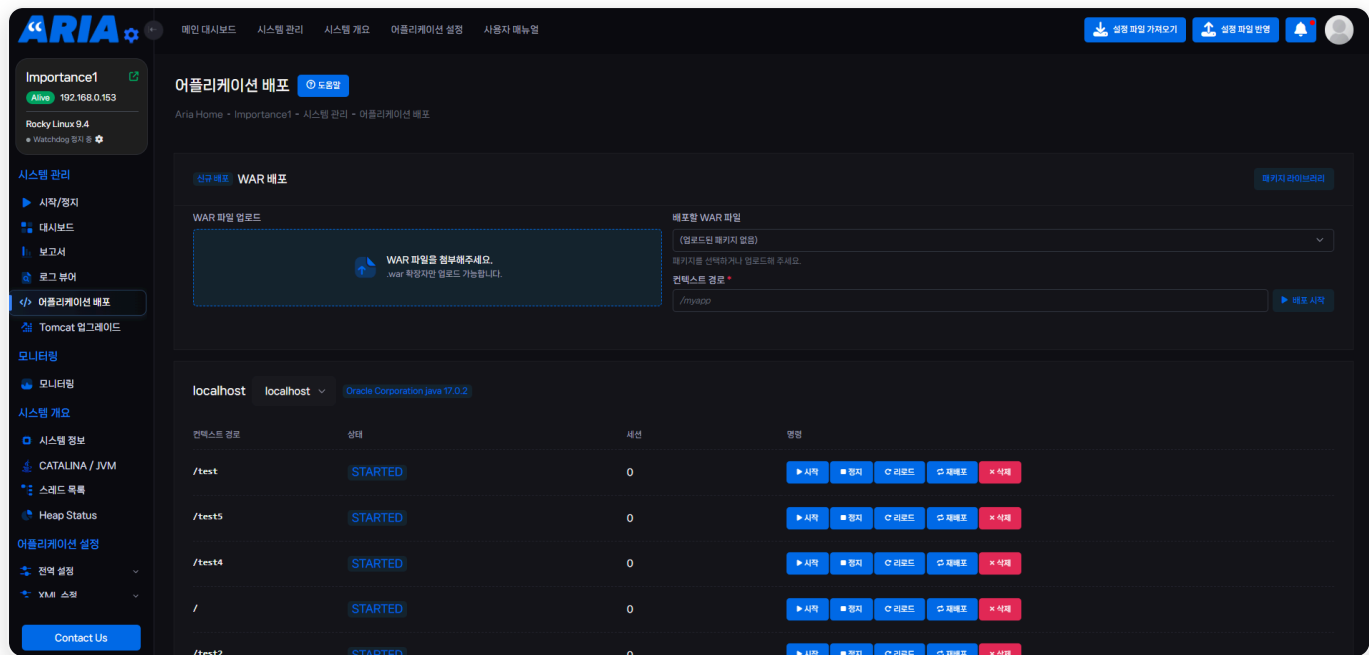
- **장애 초기 대응:** 장애 발생 시 → `catalina` 탭 선택 → `Exception` 검색 → 오류 발생 시각 파악 → 원인 분석
- **성능 저하 분석:** Access Log 상세 분석으로 느린 API 엔드포인트를 찾아내고 최적화 대상을 선정하세요.
- **정기 검토:** 주 1회 Access Log를 다운로드하여 비정상적인 접근 패턴(크롤러, 공격 시도 등)을 확인하세요.

6.5 어플리케이션 배포

경로: `/servers/management/deploy?agent_id={agent_id}`

Tomcat 서버에 웹 애플리케이션(WAR 파일)을 원격으로 배포하고 배포 이력을 관리하는 페이지입니다. WAR 파일을 업로드하고 컨텍스트 경로를 지정한 뒤 배포할 수 있으며, 배포된 애플리케이션의 시작·정지·리로드·재배포를 웹에서 직접 제어합니다.

6.5.1 화면



6.5.2 1. 화면 구성

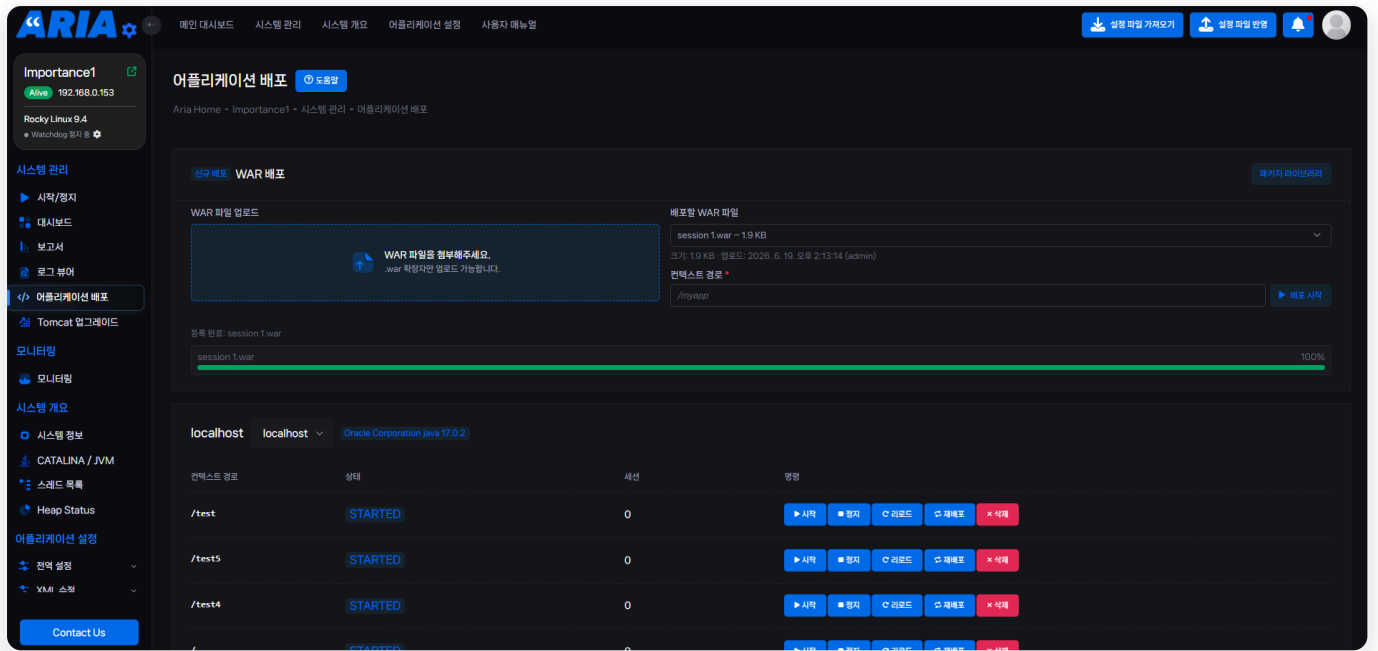
영역	설명
WAR 파일 업로드	드래그 앤 드롭 또는 클릭으로 .war 파일을 업로드
배포 설정	업로드한 WAR 파일 선택, 컨텍스트 경로 입력, 배포 시작
배포 목록	현재 배포된 애플리케이션 목록 및 상태
패키지 라이브러리	서버에 저장된 기존 WAR 파일 목록 조회·선택 (선택 사항)

6.5.3 2. WAR 파일 업로드 및 배포

신규 배포는 아래 순서로 진행합니다. **패키지 라이브러리를 거치지 않아도** 바로 배포할 수 있습니다.

WAR 파일 업로드

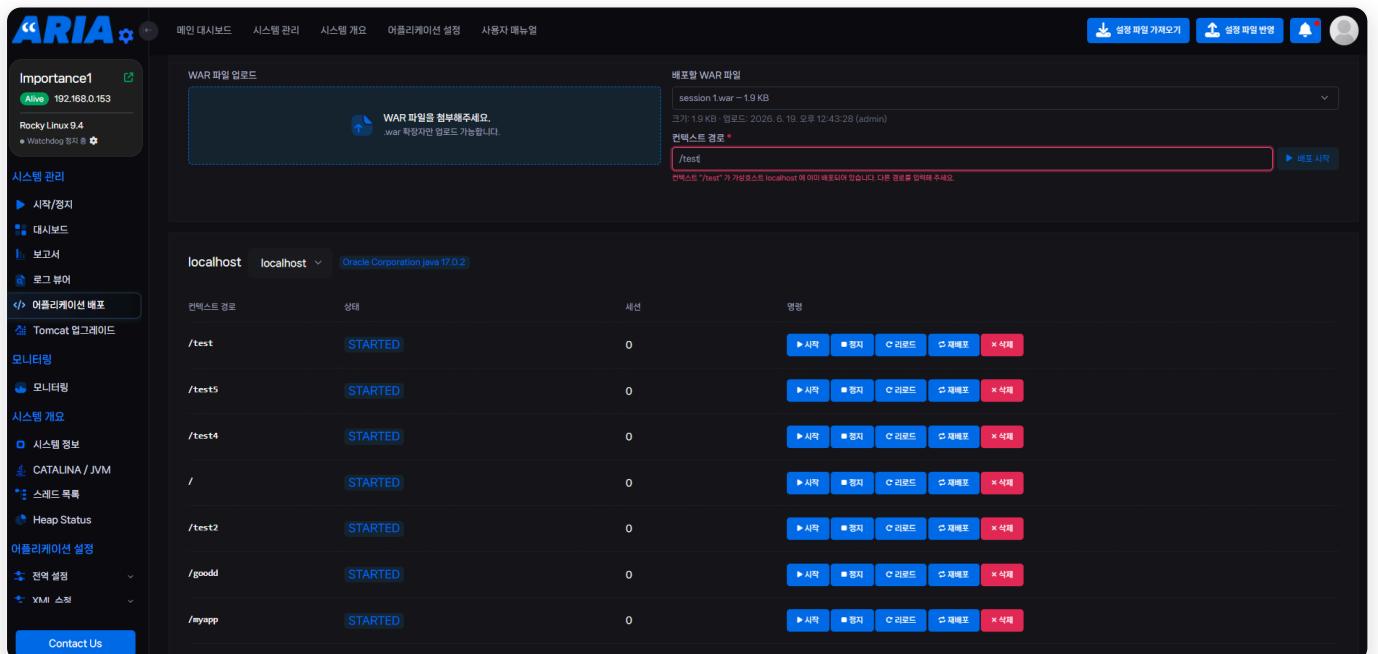
1. **WAR 파일 업로드** 영역에 .war 파일을 드래그 앤 드롭하거나, 영역을 클릭해 파일 선택 창에서 업로드합니다.
2. 업로드가 완료되면 오른쪽 **배포할 WAR 파일** 드롭다운에 파일이 자동으로 선택되고, 파일 크기·업로드 일시 등 메타 정보가 표시됩니다.



.war 확장자만 업로드할 수 있습니다.

컨텍스트 경로 입력

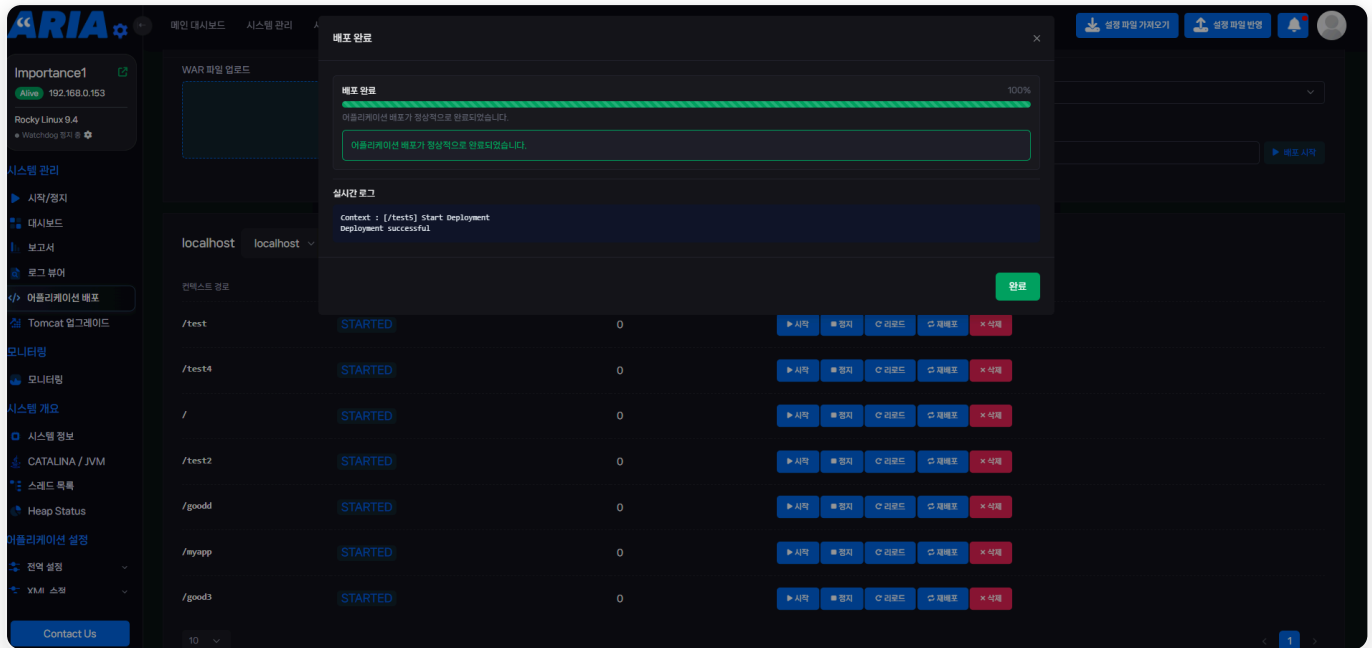
1. 컨텍스트 경로 필드에 배포할 URL 경로를 입력합니다. (예: /myapp)
2. 하단 배포 목록에 이미 존재하는 경로와 중복되면 입력 필드 아래에 경고 메시지가 표시됩니다. 중복된 경로로는 배포를 시작할 수 없으므로 다른 경로를 입력하세요.



배포 시작

1. 배포할 WAR 파일과 컨텍스트 경로를 확인합니다.
2. 배포 시작 버튼을 클릭합니다.

3. 배포 진행 상태가 실시간으로 표시됩니다.



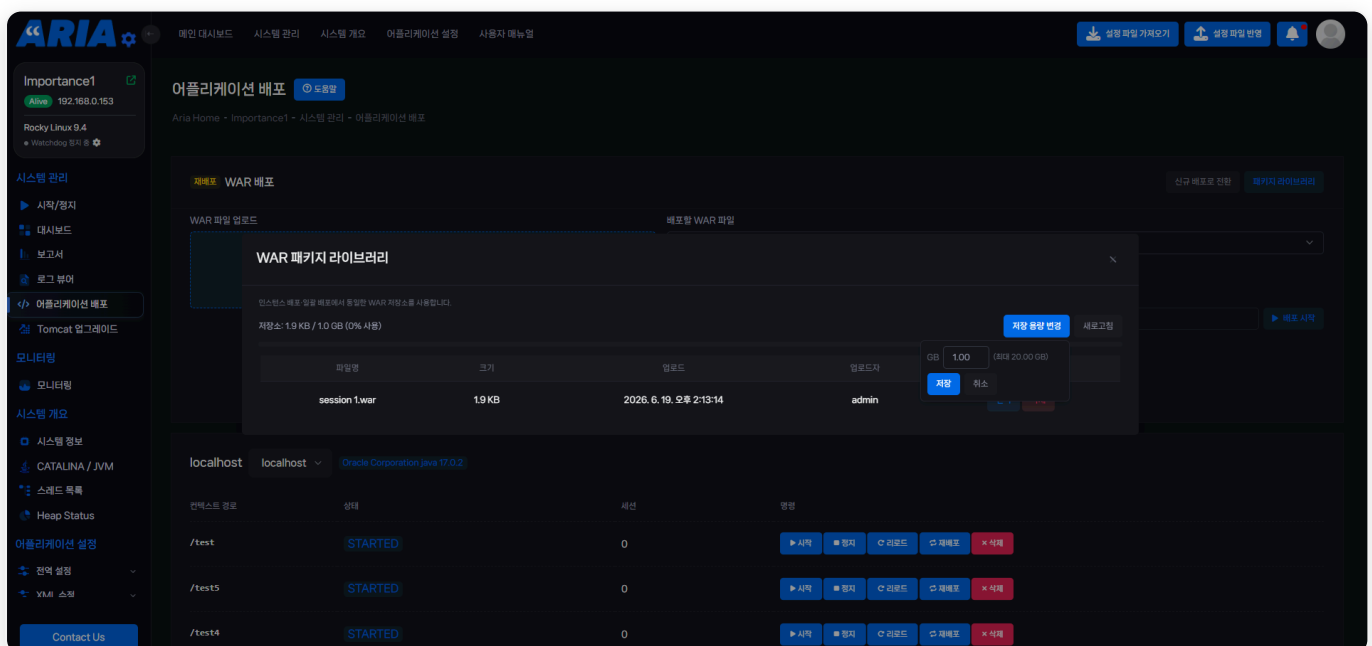
주의: 배포 중에는 기존 애플리케이션이 일시 중단됩니다. 무중단 배포가 필요하다면 로드 밸런서 설정을 검토하세요.

6.5.4 3. 패키지 라이브러리

패키지 라이브러리는 배포 절차의 필수 단계가 아닙니다. WAR 파일을 업로드하면 ARIA Server에 저장되며, 이후 **패키지 라이브러리**에서 기존에 업로드한 파일 목록을 확인하거나 다시 선택할 수 있습니다.

로컬 PC에서 같은 파일을 반복 업로드하지 않고 이전에 올린 WAR를 재사용할 때 활용합니다.

1. **패키지 라이브러리** 버튼을 클릭합니다.



1. 라이브러리 목록에서 WAR 파일을 선택하면 **배포할 WAR 파일** 드롭다운에 반영됩니다.
2. 컨텍스트 경로를 입력한 뒤 **배포 시작**을 클릭합니다.

6.5.5 4. 배포된 애플리케이션 관리

배포 목록에서 각 애플리케이션의 상태를 제어합니다.

시작 / 정지

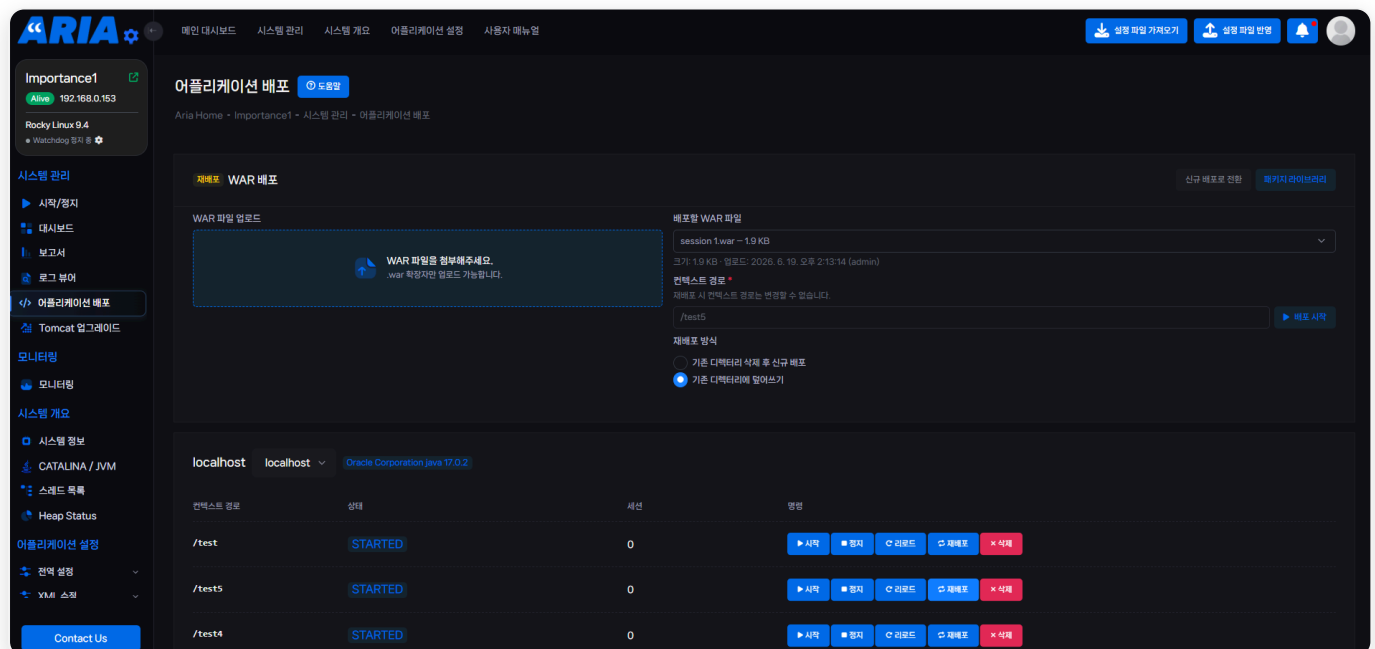
버튼	동작
시작	중지된 애플리케이션을 시작합니다
정지	실행 중인 애플리케이션을 정지합니다

리로드

애플리케이션을 재시작하지 않고 설정 파일(web.xml 등)을 다시 읽어들이는 작업입니다. 설정 변경 후 전체 재배포 없이 빠르게 반영할 때 사용합니다.

재배포

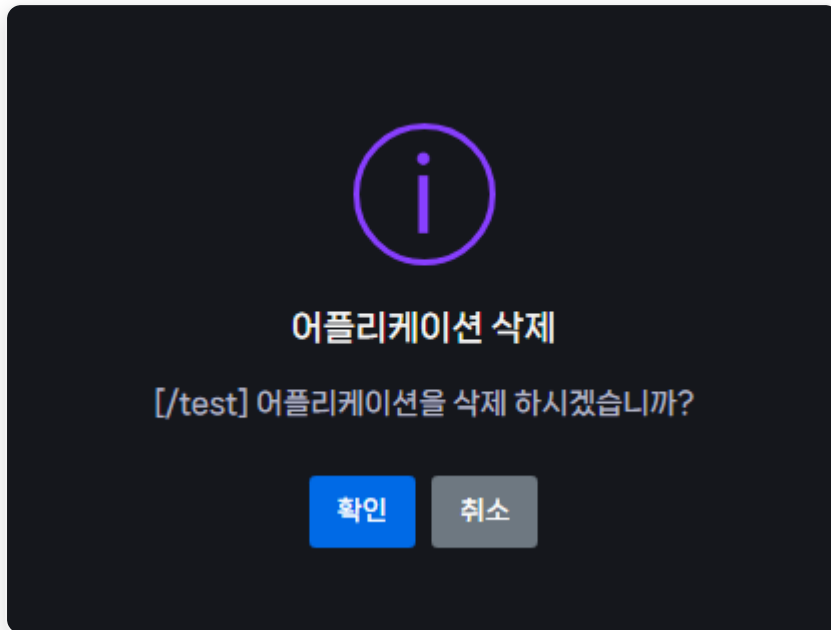
현재 배포된 WAR 파일로 애플리케이션을 다시 배포합니다. 코드 변경 없이 초기화나 메모리 클리어가 필요할 때 사용합니다.



삭제

배포된 애플리케이션을 Tomcat에서 제거합니다.

1. 삭제할 애플리케이션의 **삭제** 버튼을 클릭합니다.
2. 확인 모달에서 삭제를 확정합니다.



주의: 삭제하면 해당 Context Path의 서비스가 즉시 중단됩니다.

6.5.6 💡 활용 팁

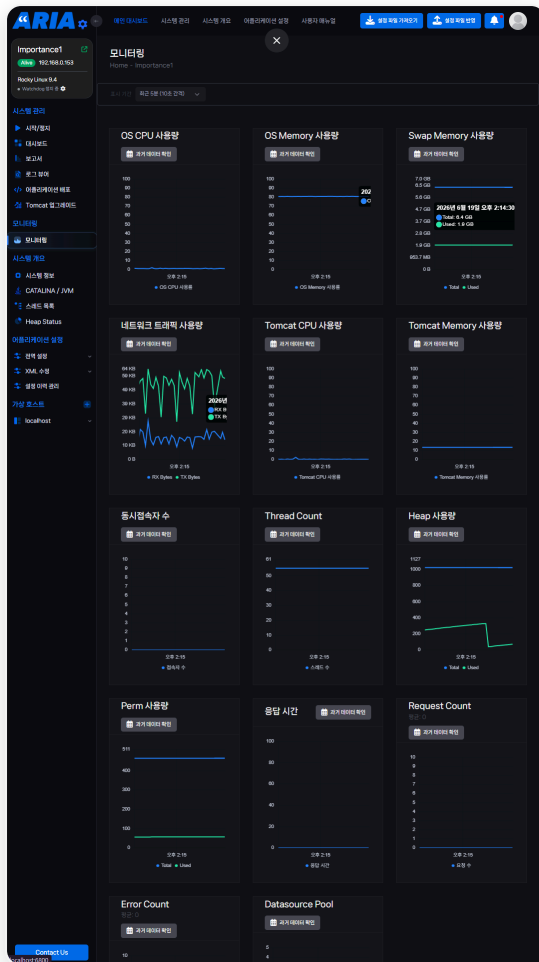
- **배포 전 확인:** 배포 전 [로그 뷰어](#)를 Tail 모드로 열어두면 배포 과정에서 발생하는 에러를 실시간으로 확인할 수 있습니다.
- **롤백:** 배포 후 문제가 발생하면 이전 버전 WAR 파일로 재배포하여 빠르게 롤백하세요.

6.6 모니터링

경로: `/servers/management/monitoring?agent_id={agent_id}`

선택한 Tomcat 서버의 성능 지표(CPU/메모리, 스레드 사용률, 활성 세션 수, RPS, 에러율, 평균 응답 시간, DataSource 연결 등)를 실시간 시계열 차트로 종합 모니터링하는 페이지입니다.

6.6.1 화면



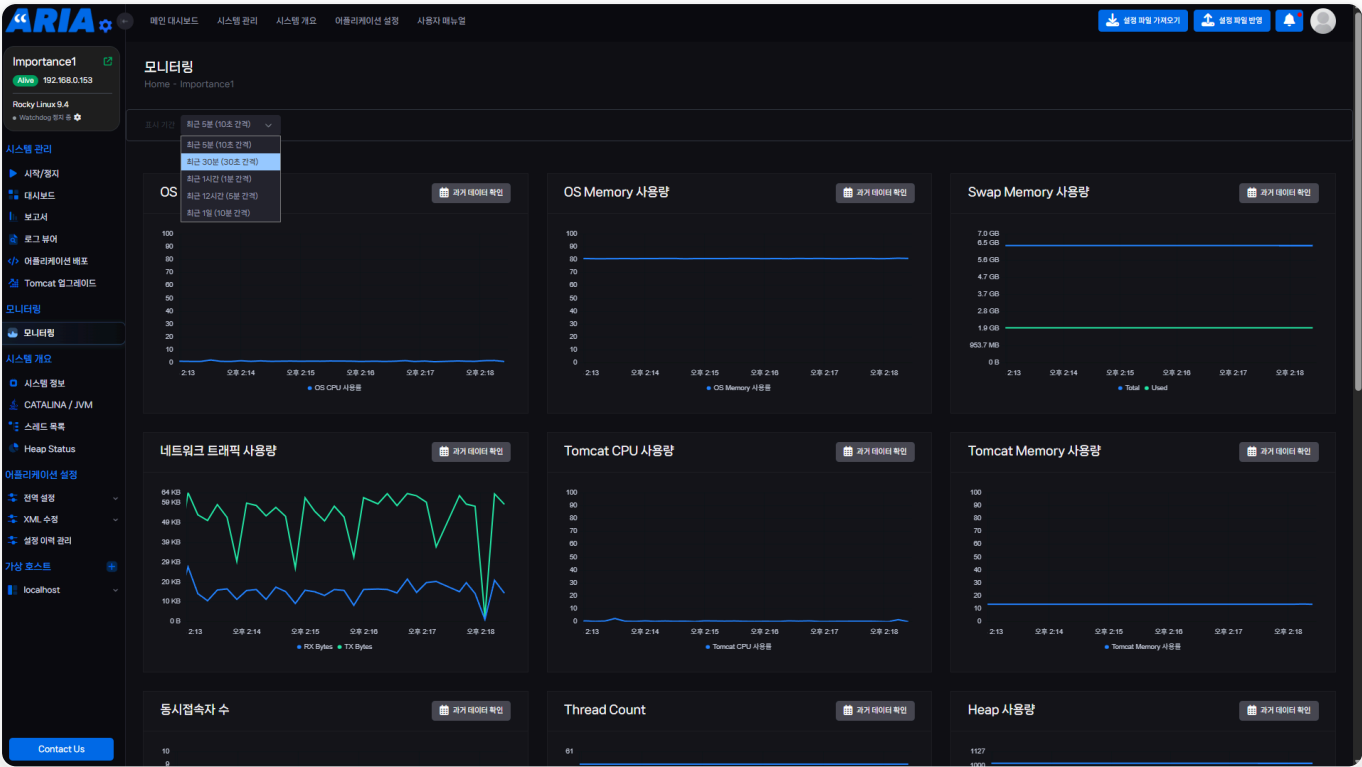
6.6.2 1. 실시간 모니터링 메인 화면

1-1. 모니터링 지표 구성

- **OS CPU / Memory 사용률**: 하드웨어 인프라 리소스 소요량 확인
- **Tomcat CPU / Memory 사용률**: 톰캣 WAS 인스턴스의 순수 자원 소요량 확인
- **JVM Heap / Non-Heap 사용량**: 가비지 컬렉션(GC) 튜닝의 기준이 되는 힙 메모리 및 메타스페이스 사용률
- **Tomcat 스레드 수**: 현재 사용 중인 액티브 스레드와 최대 풀 제한량 확인
- **동시 사용자 수 (Session)**: 활성화된 HTTP 동시 접속 세션 수
- **평균 응답 시간 / 요청 수 / 에러 수**: WAS의 성능을 결정하는 3대 주요 트래픽 지표
- **DataSource (JDBC DB 연결)**: 각 데이터베이스 연결 풀(Active/Idle 연결 수) 실시간 모니터링 (단, Tomcat 기본 컨텍스트 `/manager`, `/docs` 등은 필터링되어 제외됨)

1-2. 시간 범위 (Scope) 제어

우상단 시간 범위 드롭다운 옵션을 통해 대시보드 전체 차트의 조회 범위를 제어합니다.



- 옵션: 5분, 30분, 1시간, 6시간, 1일 (지정된 범위에 따라 실시간 갱신 주기가 결정됨)

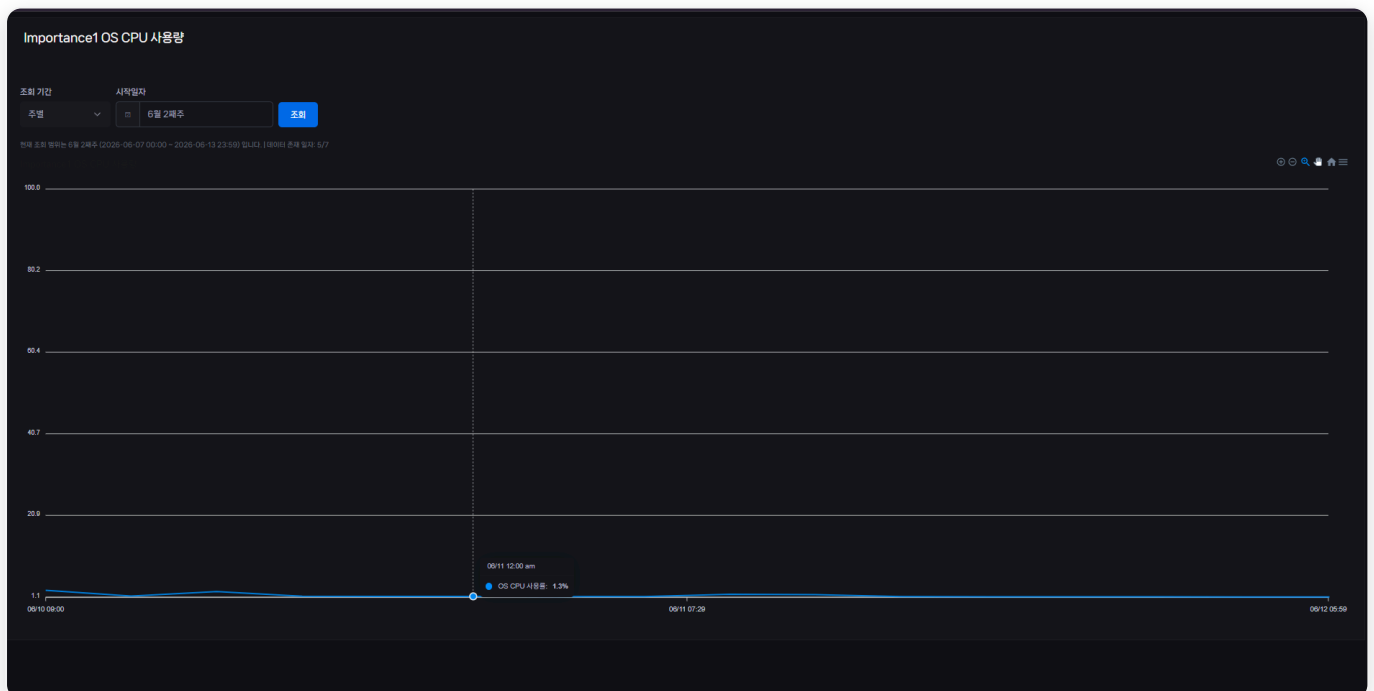
1-3. 범례 (Legend) 활용 필터링

차트 하단 범례의 항목을 클릭하여 특정 데이터를 차트에서 일시적으로 숨기거나 노출시킬 수 있습니다.



2. 개별 지표 상세 모니터링 모달

각 차트 영역 우상단의 **확대** 아이콘을 클릭하면 세부 데이터 조회를 위한 모달 창이 노출됩니다. 상세 모달에서는 보고서 생성 모듈에 준하는 다차원 시점 조회(Scope)와 집계 데이터 가공을 지원합니다.



2-1. 상세 조회 범위(Scope) 및 집계 간격

조회하려는 모드에 따라 차트에 출력되는 데이터 포인트의 **집계 간격(Interval)**이 동적으로 조절되어 최적의 속도를 유지합니다.

상세 범위 (Scope)	설명	데이터 집계 간격
일간 (Daily)	지정한 특정 1일 범위 내 시간대별 추이 조회	원시 데이터 그대로 (집계 없음)
주간 (Weekly)	지정한 특정 1주일(7일) 범위 내 추이 조회	3시간 평균값으로 집계 가공
월간 (Monthly)	지정한 특정 1개월(30일) 범위 내 추이 조회	1일 평균값으로 집계 가공
사용자 정의 (Custom)	직접 시작일~종료일을 설정한 기간 조회	5분 평균값으로 집계 가공

(각 상세 범위별 설정 예시)

일간 (Daily)

Importance1 OS CPU 사용량

조회 기간

일별

▼

시작일자

2026-06-19

⊞

시작 시간

12:00 AM

⊞

종료 시간

11:59 PM

⊞

조회

현재 조회 범위는 2026-06-19 00:00 -

<

26년 6월

>

일	월	화	수	목	금	토
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	1	2	3	4
5	6	7	8	9	10	11

📅

주간 (Weekly)

Importance1 OS CPU 사용량

조회 기간

주별

▼

시작일자

6월 3째주

⊞

조회

현재 조회 범위는 6월 3째주 (2026-06-1

<

26년 6월

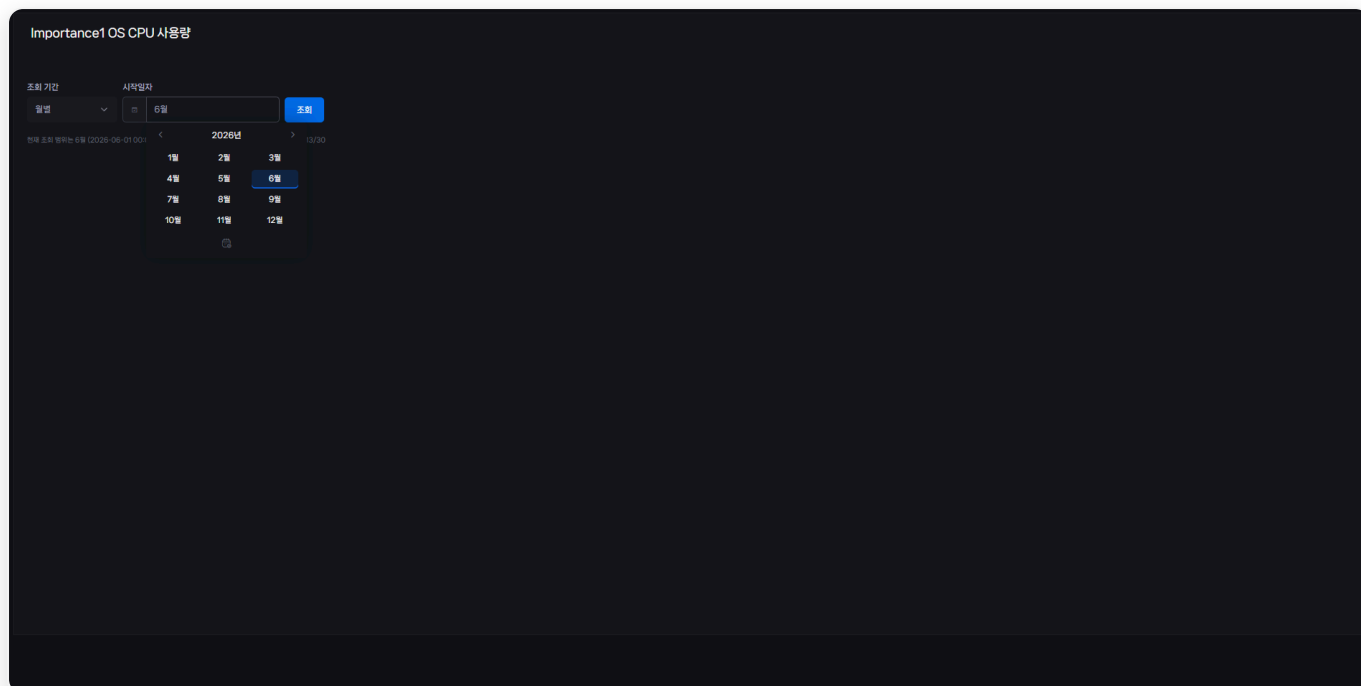
>

일	월	화	수	목	금	토
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	1	2	3	4
5	6	7	8	9	10	11

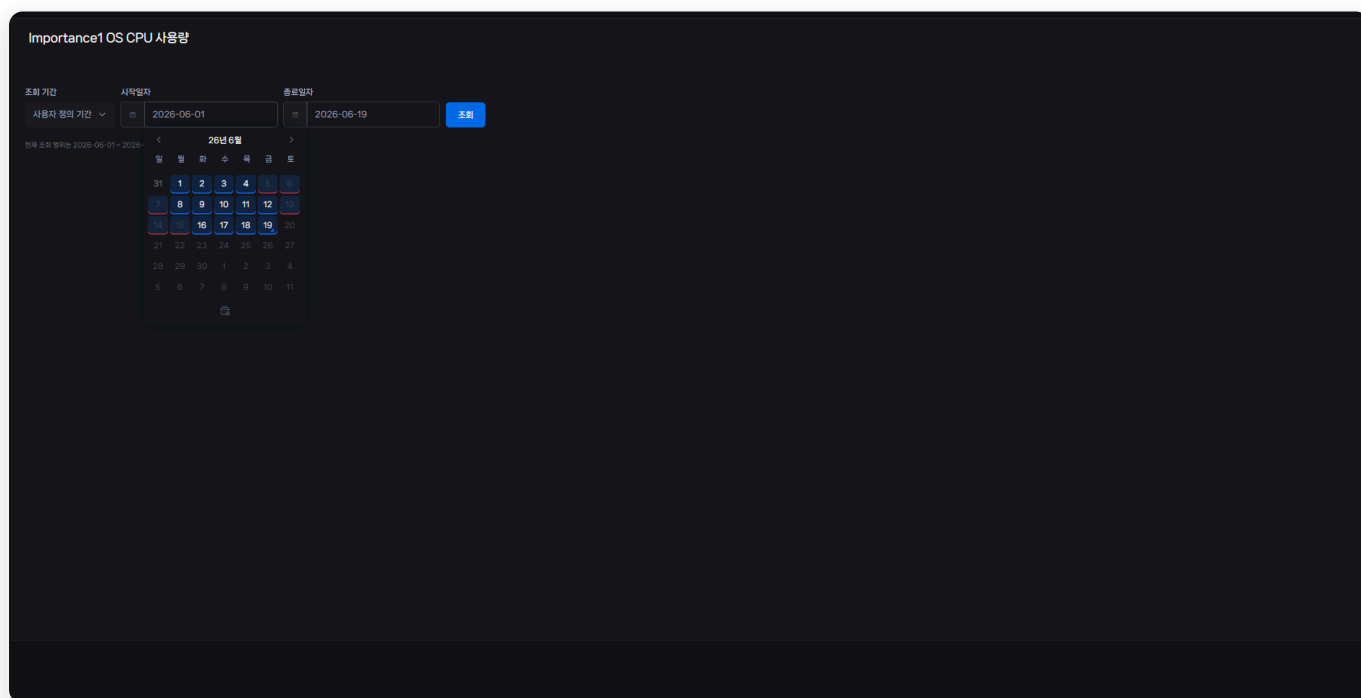
📅

일차: 4/7

월간 (Monthly)



사용자 정의 (Custom)



2-2. 캘린더 제약 조건 및 데코레이션

모달 내 날짜 선택기는 서버로부터 **데이터가 수집된 날짜 목록**을 받아와 자동으로 필터링 및 시각화 처리를 수행합니다.

- **오늘 날짜 항상 활성화:** 데이터 유무와 상관없이 오늘 날짜는 항상 선택 가능하도록 처리됩니다.
- **월간 캘린더 연계:** 월간(Month) 선택 시에는 해당 달에 데이터가 하루라도 존재하면 그 달의 전체 일자가 활성화됩니다.
- **캘린더 표시 구분:**
- **선택된 기간:** 현재 탐색 범위로 선택된 날짜

- **데이터 있음:** 실제 데이터가 적재되어 조회 가능한 일자
- **데이터 없음:** 데이터가 존재하지 않아 비활성화된 일자 (마우스 오버 시 "데이터 없음" 툴팁 표시)

2-3. 일간 시간 유효성 체크

일간(**Daily**) 조회 설정 시, 관리자가 시작 시간과 종료 시간을 직접 입력하여 조회할 수 있습니다.

- **검증:** 종료 시간이 시작 시간보다 같거나 이른 시간으로 설정된 상태에서 조회를 시도하면 브라우저 유효성 검증 경고가 발생하며 조회가 제한됩니다.

6.6.3 💡 활용 팁

- **임계치 대응:** RPS(초당 요청 수)나 Active 스레드가 급격히 이상향하는 추세가 상세 모니터링 주간/월간 차트에서 관측된다면 인프라 증설 및 부하 분산을 검토하십시오.
- **성능 하락 상관분석:** 특정 응답 시간 저하 현상이 감지될 시, 상세 모달을 통해 CPU 사용률 차트와 Tomcat 메모리 차트의 피크 타임을 비교 분석하여 물리 리소스 부족 또는 어플리케이션 병목 여부를 진단하십시오.

6.7 시스템 정보

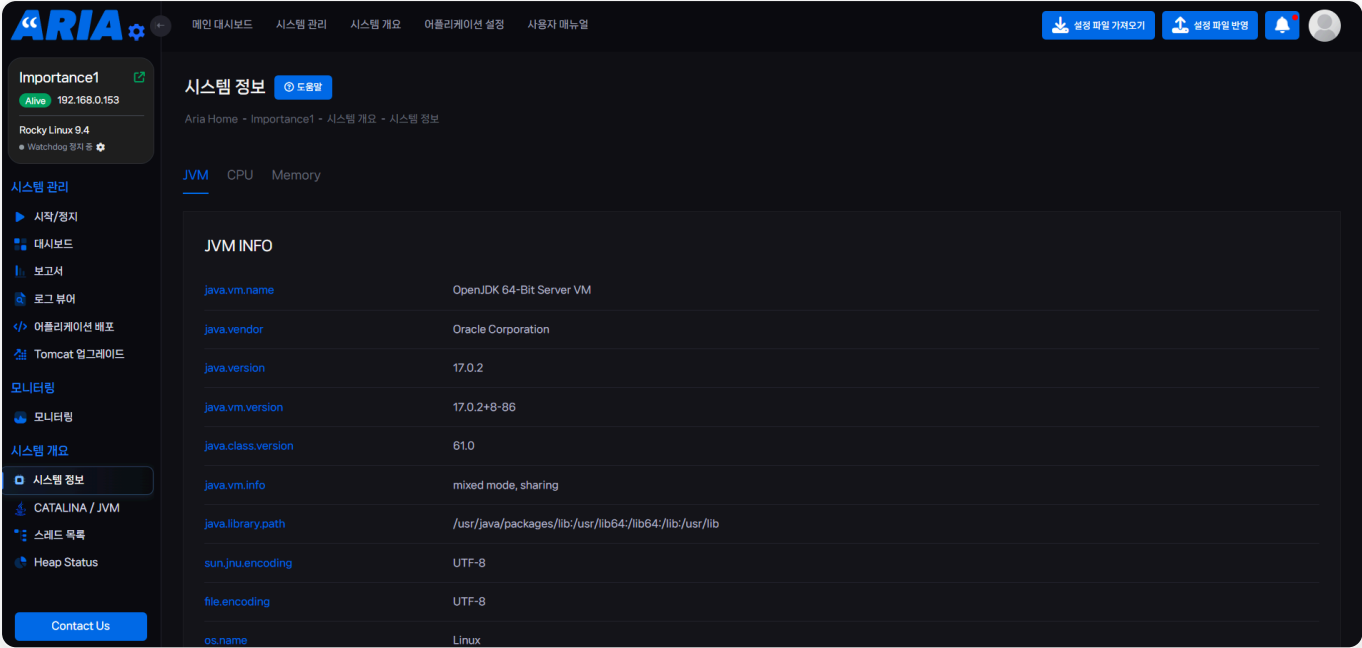
경로: `/servers/management/systemInfo?agent_id={agent_id}`

ARIA 에이전트가 설치된 대상 서버의 물리적인 하드웨어(CPU, Memory) 사양 및 Java 실행 환경(JVM) 시스템 프로퍼티 정보를 상세하게 조회하는 페이지입니다.

[!!IMPORTANT] **서버 기동 상태 필수:** 대상 Tomcat 서버가 기동 중(**Alive**)인 상태에서만 시스템 정보의 실시간 조회가 가능합니다. 서버가 중지(**Dead**) 상태일 경우 "서버가 구동중이 아닙니다." 라는 안내와 함께 정보 출력이 제한됩니다.

6.7.1 1. JVM (Java Virtual Machine) 정보

Tomcat WAS가 동작 중인 JVM의 구동 사양과 핵심 시스템 프로퍼티 값을 확인합니다.

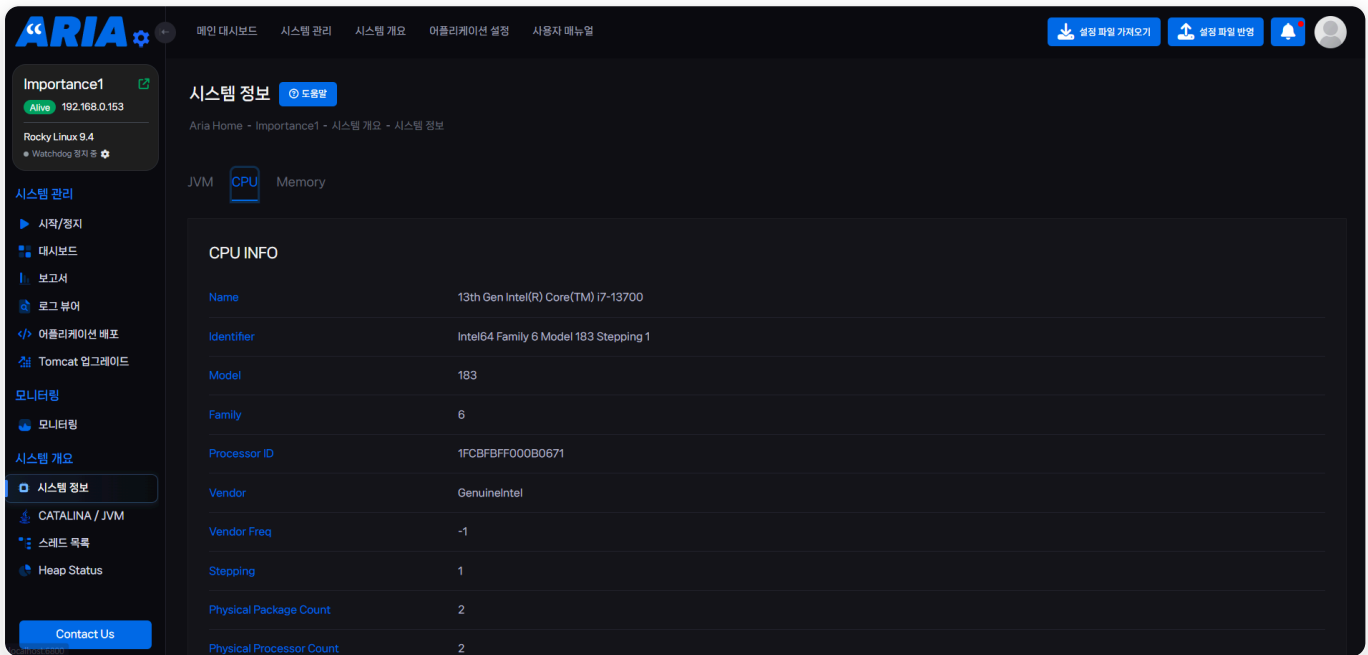


설정 키 (Property)	설명
java.vm.name	Java 가상 머신(JVM)의 정식 명칭
java.vendor	Java Runtime Environment(JRE) 배포 및 제공사 정보
java.version	구동 중인 Java의 릴리즈 버전
java.vm.version	JVM 가상 머신의 세부 릴리즈 버전
java.class.version	Java 클래스 파일의 버전
java.vm.info	JVM 가상 머신의 작동 모드 및 컴파일 방식 정보
java.library.path	Java 네이티브 라이브러리(.dll, .so) 참조 경로 목록
sun.jnu.encoding	운영체제 시스템 기본 문자 인코딩 설정
file.encoding	파일의 기본 문자 인코딩
os.name	대상 서버의 운영체제(OS) 제품명
os.arch	대상 서버의 하드웨어/OS 비트 아키텍처 (예: amd64, x86)
user.name	JVM 프로세스를 실행한 서버 계정 소유자 이름

설정 키 (Property)	설명
user.language	프로세스 로케일의 기본 설정 언어
user.timezone	JVM 환경에서 시간 계산에 사용하는 표준 시간대

6.7.2 2. CPU 정보

대상 서버에 장착된 물리 프로세서의 아키텍처와 프로세서 개수 등의 세부 하드웨어 정보를 파악합니다.

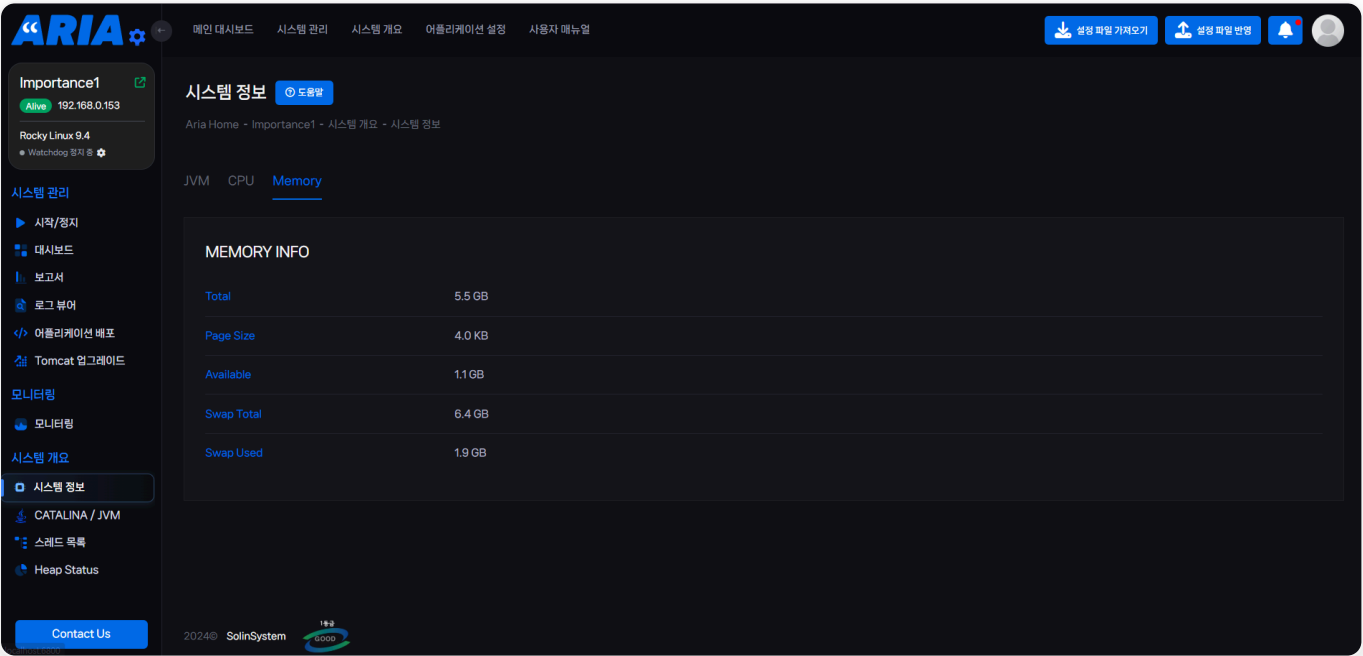


항목 (Item)	설명
Name	CPU의 공식 프로세서 제품명 (예: Intel Core / AMD Ryzen 등)
Identifier	하드웨어 레벨의 CPU 고유 식별자 문자열
Model	제조업체에서 부여한 CPU 모델 번호
Family	프로세서 세대 및 제품 계열 번호
Processor ID	하드웨어 프로세서 식별 아이디
Vendor	CPU 칩셋 제조업체 브랜드 (예: GenuineIntel, AuthenticAMD)
Vendor Freq	CPU 제조업체 기준 기본 클럭 속도 (Hz)

항목 (Item)	설명
Stepping	CPU 제조 프로세스의 미세 공정 개선 버전(스테핑) 단계
Physical Package Count	메인보드에 실제로 장착된 물리 CPU 소켓의 총 개수
Physical Processor Count	CPU 내부에 하드웨어적으로 탑재된 물리적인 코어(Core)의 총 개수
Logical Processor Count	하이퍼스레딩 등으로 인식되는 OS 상의 논리적 프로세서(스레드) 총 개수

6.7.3 3. Memory 정보

서버의 전체 물리 메모리 용량과 스왑 공간의 점유 현황을 조회합니다.



항목 (Item)	설명
Total	장착된 전체 물리 RAM 메모리 크기 (Byte / GB)
Page Size	운영체제 가상 메모리 시스템의 기본 단위 페이지 크기
Available	OS 커널 및 다른 프로세스에서 즉시 가용 가능한 여유 메모리 크기
Swap Total	하드디스크의 가상 공간을 활용하는 전체 스왑(Swap) 메모리 크기
Swap Used	현재 사용 중인 가왓(Swap) 메모리 크기

6.7.4 💡 활용 팁

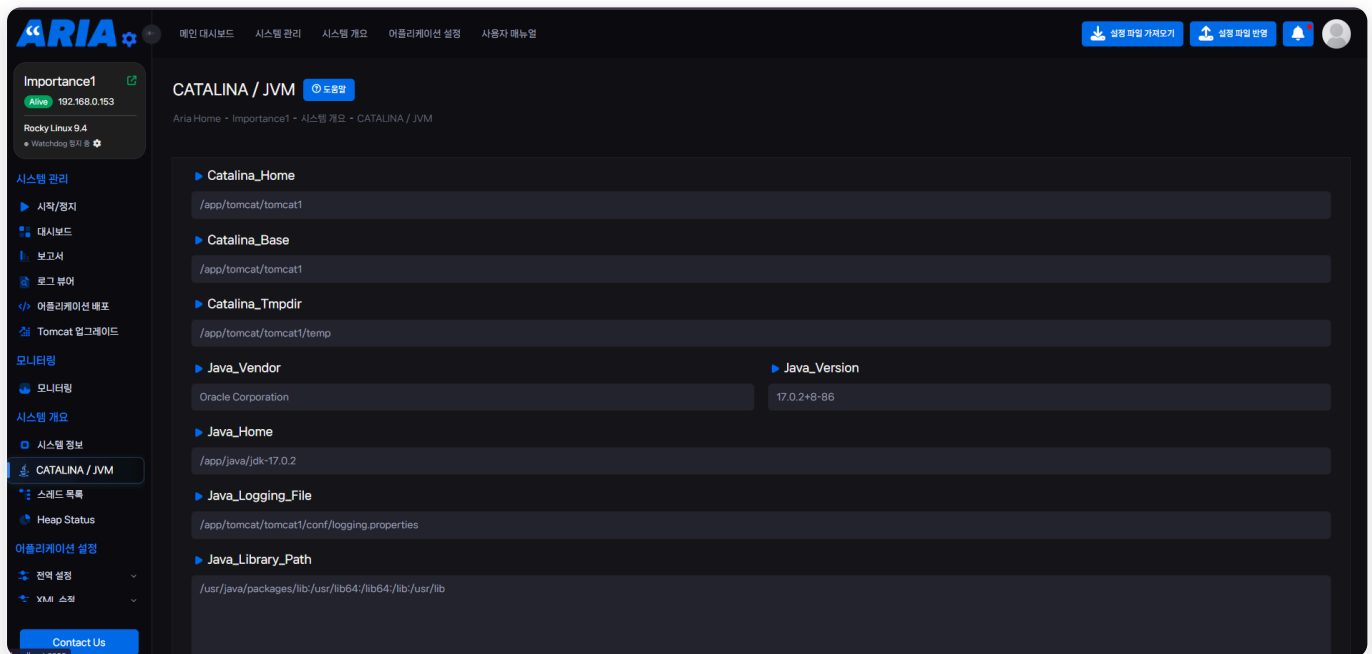
- **성능 분석 대조:** [모니터링](#) 페이지에서 CPU 사용률이 비정상적으로 치솟을 때, 해당 서버의 **Logical Processor Count**를 확인하여 병렬 처리 설계 한계치를 파악할 수 있습니다.
- **인코딩 충돌 해결:** 한글 깨짐이나 시스템 파일 입출력 에러 발생 시, JVM 탭에서 **file.encoding** 설정을 확인하여 한글 지원 표준 규격(UTF-8 등)과 맞는지 대조하세요.

6.8 CATALINA / JVM

경로: `/servers/management/jvm?agent_id={agent_id}`

Tomcat의 Catalina 설치 경로와 Java 실행 환경 정보를 조회하고, `setenv.sh` (또는 `setenv.bat`) 파일을 통해 JVM 메모리·GC 옵션 등을 편집·반영하는 페이지입니다.

6.8.1 화면



6.8.2 1. 화면 구성

영역	설명
Catalina / Java 정보	Tomcat 경로 및 Java 버전 등 읽기 전용 메타 정보
JVM 옵션 편집기	<code>setenv.sh</code> / <code>setenv.bat</code> (또는 윈도우 서비스 Java Options) 내용 편집

영역	설명
반영	편집한 JVM 옵션을 서버에 저장

내장형 Tomcat: 내장형 Tomcat 서버는 JVM 옵션 편집기와 **반영** 버튼이 표시되지 않습니다.

6.8.3 2. Catalina / Java 메타 정보

에이전트가 수집한 Tomcat JVM 환경 값을 읽기 전용으로 표시합니다.

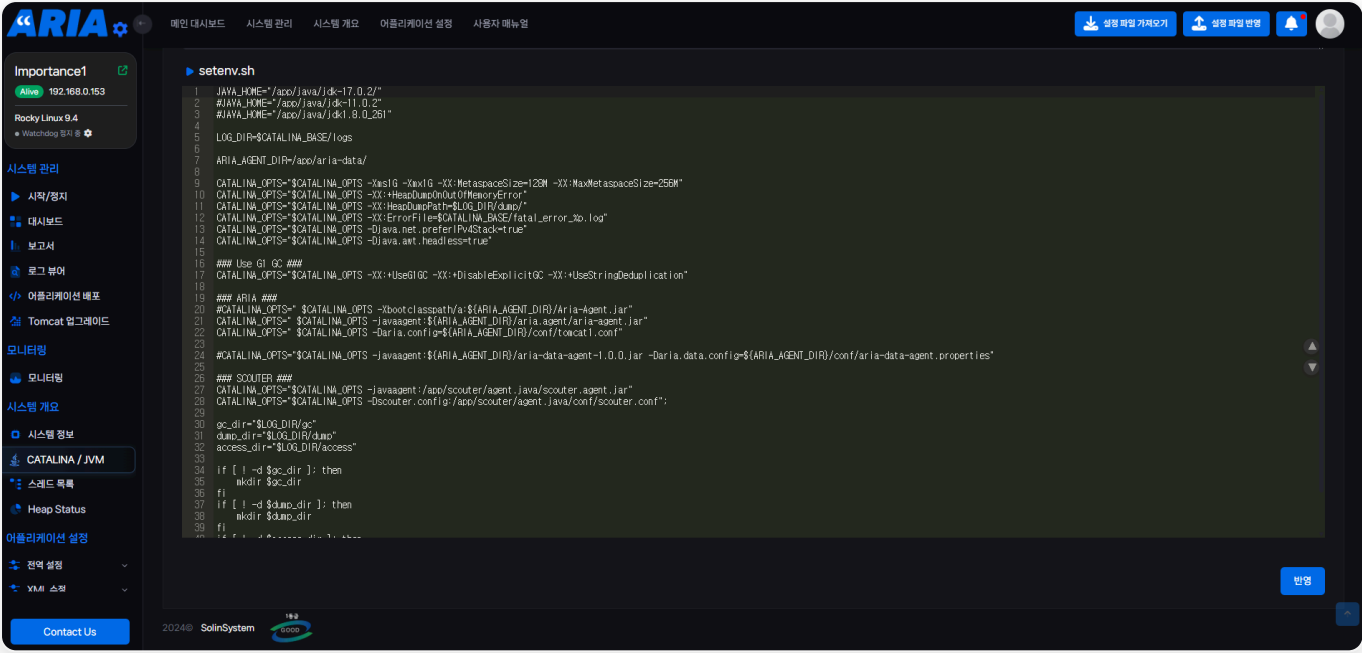
항목	설명
Catalina_Home	Tomcat 기본 설치 디렉터리
Catalina_Base	Tomcat 런타임 디렉터리
Catalina_Tmpdir	Tomcat 임시 파일 디렉터리
Java_Vendor	Java 배포사
Java_Version	Java 릴리스 버전
Java_Home	Java 설치 디렉터리
Java_Logging_File	Java 로깅 설정 파일 경로
Java_Library_Path	Java 네이티브 라이브러리 참조 경로

6.8.4 3. JVM 옵션 편집

운영체제와 Tomcat 등록 방식에 따라 편집 대상 파일이 달라집니다.

환경	편집 대상
Linux	<code>setenv.sh</code>
Windows (일반)	<code>setenv.bat</code>
Windows (서비스 등록)	서비스명 Java Options (옵션마다 줄바꿈)

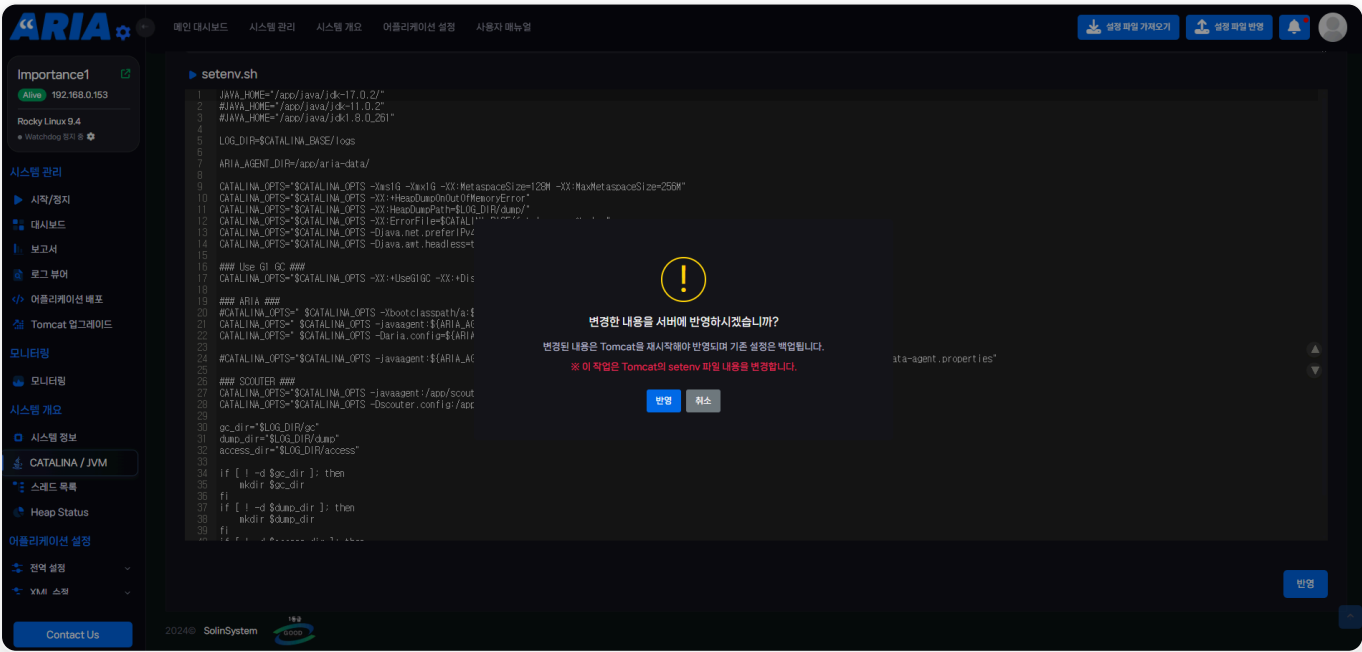
페이지 로드 시 에이전트에서 현재 설정을 불러와 에디터에 표시합니다. 힙 크기(`-Xms` , `-Xmx`), GC 알고리즘(`-XX:+UseG1GC` 등), 에이전트 옵션 등을 이 영역에서 수정합니다.



6.8.5 4. 설정 반영

편집한 JVM 옵션을 Tomcat 서버에 저장합니다.

1. 에디터에서 JVM 옵션을 수정합니다.
2. 하단 **반영** 버튼을 클릭합니다.
3. 확인 모달에서 반영을 확정합니다.



버튼	동작
반영	<code>setenv</code> 파일(또는 윈도우 서비스 Java Options)에 내용 저장
취소	저장하지 않고 모달 닫기

주의: 변경 내용은 **Tomcat 재시작** 후에 적용됩니다. 반영 시 기존 설정 파일은 백업됩니다. 원격 쓰기가 제한된 에이전트는 반영 전에 추가 확인이 필요할 수 있습니다.

반영이 완료되면 "설정 파일이 정상적으로 수정되었습니다." 안내가 표시됩니다.

6.8.6 💡 활용 팁

- **JVM 메모리 모니터링:** 힙·Non-Heap 사용량 추이는 [모니터링](#) 페이지에서 확인하세요. 이 페이지는 설정 편집 전용이며 실시간 GC 통계 차트는 제공하지 않습니다.
- **메모리 누수 분석:** OOM이나 메모리 누수가 의심되면 [힙 덤프](#)를 생성해 분석하세요.
- **GC 튜닝:** `-Xmx` 확대, G1 GC 적용 등은 `setenv` 편집기에서 `CATALINA_OPTS`에 옵션을 추가한 뒤 반영하고 Tomcat을 재시작하세요.

6.9 스레드

경로: `/servers/management/thread?agent_id={agent_id}`

Tomcat JVM에서 현재 실행 중인 모든 스레드의 실시간 상태를 모니터링하고 분석하는 페이지입니다. 서버 응답 지연, CPU 점유율 급증, 또는 데드락(Deadlock)이 의심되는 상황에서 개별 스레드의 Stack Trace를 실시간 조회하거나 전체 Thread Dump를 생성하여 정확하게 원인을 진단할 수 있습니다.

The screenshot shows the ARIA monitoring interface. The main content area is titled "스레드 목록" (Thread List) and displays a table of threads. The table has columns for Thread Name, Thread ID, State, Priority, and Stack. The threads listed are:

THREAD NAME	THREAD ID	STATE	PRIORITY	STACK
main	1	RUNNABLE	5	[stack trace]
Reference Handler	2	RUNNABLE	10	[stack trace]
Finalizer	3	WAITING	8	[stack trace]
Signal Dispatcher	4	RUNNABLE	9	[stack trace]
Common-Cleaner	10	TIMED_WAITING	8	[stack trace]
FileHandlerLogFilesCleaner-1	13	WAITING	5	[stack trace]
Configure	14	TIMED_WAITING	5	[stack trace]
Aria-Log	15	TIMED_WAITING	5	[stack trace]
Scouter-Configure	17	TIMED_WAITING	5	[stack trace]
Scouter-BackJobs	21	TIMED_WAITING	5	[stack trace]

The interface also includes a sidebar with navigation options like System Overview, Monitoring, and Settings, and a top bar with user information and system status.

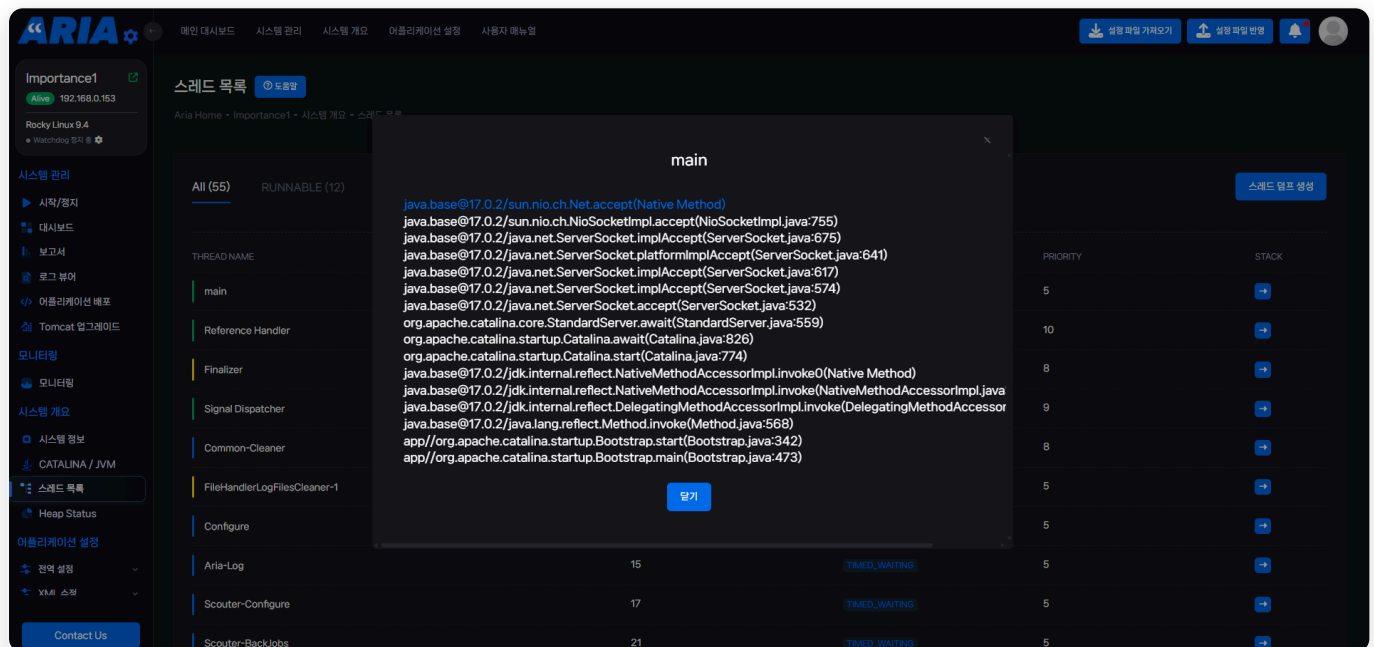
6.9.1 1. 주요 기능 및 사용법

1) 스레드 상태 필터링 탭

스레드 목록 상단의 탭 영역을 통해 각 상태별 스레드 수를 한눈에 파악하고, 원하는 상태의 스레드만 실시간으로 필터링하여 조회할 수 있습니다. * **지원 상태 탭:** ALL, RUNNABLE, WAITING, TIMED_WAITING, BLOCKED, DEADLOCKED, NEW, TERMINATED * **미구동 안내:** 대상 Tomcat 서버가 정상적으로 기동되지 않았거나 연결이 불가능한 경우, 테이블에 Tomcat 서버가 구동중이 아닙니다. 라는 안내 메시지가 노출됩니다.

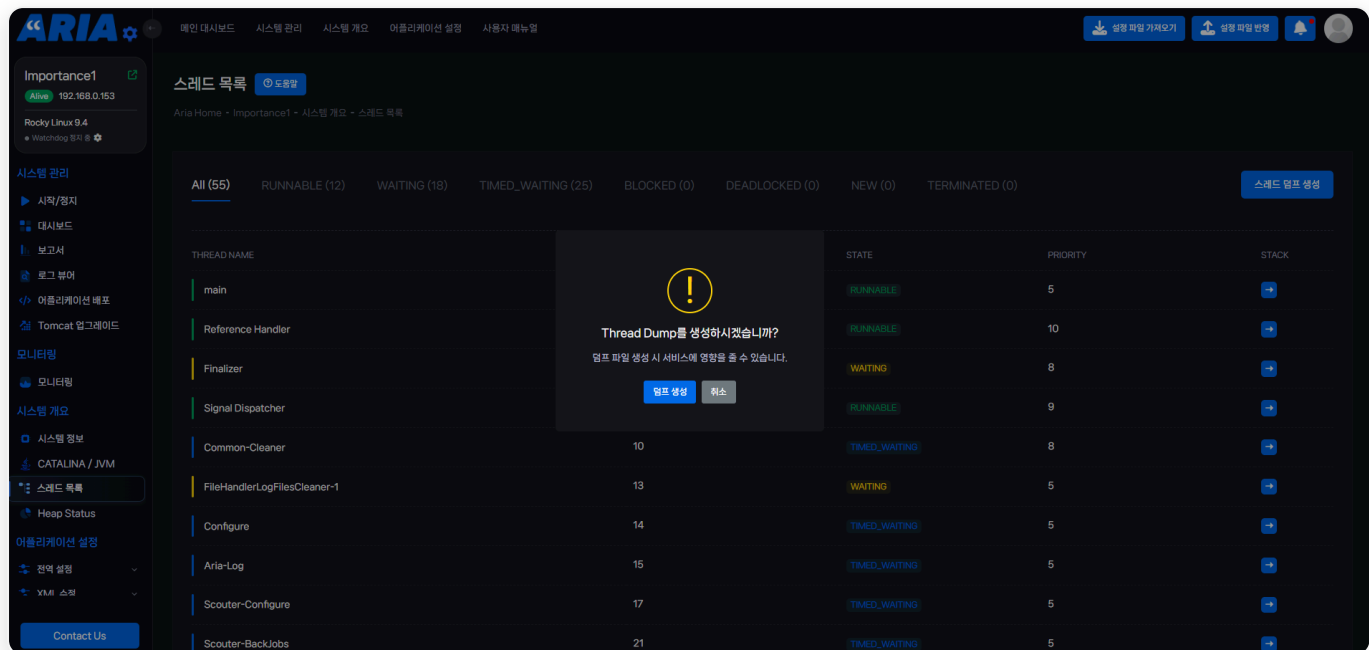
2) 실시간 Stack Trace 조회

특정 스레드의 상세 실행 경로(Stack Trace)를 실시간으로 추적하여 코드 수준의 병목 구간을 탐색합니다. * **조회 방법:** 스레드 목록 우측 끝에 위치한 실행 버튼(화살표 모양 아이콘)을 클릭합니다. * **스택 강조:** 실시간 API 호출을 통해 수집된 Stack Trace 중 가장 상위 실행 지점(Top Stack)은 별도로 구분되어 강조 표시되며, 하위 실행 경로(Remaining Stack)와 구분하여 한눈에 문제 지점을 파악할 수 있도록 모달 창에 시각적으로 표시됩니다.



3) Thread Dump 생성 및 다운로드/뷰어

서버 내 전체 스레드의 상태 스냅샷 정보가 담긴 덤프 파일을 즉각적으로 생성하여 정밀 분석을 수행할 수 있습니다. * **동작 제약 조건:** 대상 Tomcat 서버가 실행 중(Alive) 상태일 때만 생성이 가능합니다. 만약 서버 상태가 정지(Dead) 상태인 경우, 기능 실행 시 제한 에러 알림이 표시됩니다. * **덤프 생성 절차:** 1. 우측 상단의 스레드 덤프 생성 버튼을 클릭합니다. 2. "덤프 파일 생성 시 서버에 영향을 줄 수 있습니다." 경고 팝업에서 덤프 생성을 승인합니다. 3. 덤프 파일이 Aria 서버의 실행 디렉터리에 성공적으로 기록되면 생성 완료 팝업이 노출됩니다. * **실시간 화면 뷰어 연동:** 생성 완료 팝업 하단 푸터(Footer)의 "현재 화면에서 확인" 링크를 클릭하면 전체 화면(Fullscreen) 뷰어가 활성화되어 덤프 텍스트 파일의 내용을 즉시 읽을 수 있습니다. * **실시간 다운로드 진행을 표시:** 파일 생성과 동시에 백엔드 다운로드 세션 API와 연동된 파일 다운로드가 시작되며, 실시간 다운로드 전송 진행율(%)이 화면에 표시됩니다.



2. 스레드 상태 정의

상태	설명
RUNNABLE	JVM 내에서 CPU 할당을 받아 실제로 실행 중이거나 실행을 대기하고 있는 상태입니다.
WAITING	다른 스레드의 특정 조건(알림, Lock 해제 등) 충족을 무기한 대기하는 상태입니다. (<code>Object.wait()</code> , <code>Thread.join()</code> , <code>LockSupport.park()</code>)
TIMED_WAITING	지정된 제한 시간 동안 대기하는 상태입니다. (<code>Thread.sleep()</code> , <code>Object.wait(timeout)</code> , <code>Thread.join(timeout)</code> 등)
BLOCKED	모니터 락(Lock)을 획득하기 위해 대기하는 상태로, 다른 스레드가 해당 락을 점유하고 있어 대기 중인 상태입니다.
DEADLOCKED	둘 이상의 스레드가 서로 상대방이 가진 락을 획득하기 위해 무한히 대기하고 있는 교착 상태의 스레드입니다.
NEW	스레드 인스턴스는 생성되었으나 아직 실행(<code>start()</code>)되지 않은 상태입니다.
TERMINATED	실행이 완전히 완료되어 종료된 상태입니다.

6.9.2 💡 장애 진단 팁

- 스레드 락 경합 및 병목 진단:

- 응답이 지연되거나 CPU가 과도하게 상승할 시 **BLOCKED** 또는 **DEADLOCKED** 탭의 개수를 확인하십시오.
- BLOCKED 스레드가 다수 발생할 경우, 실시간 Stack Trace 조회를 이용해 해당 스레드가 어떤 Lock 오브젝트나 동기화 블록(synchronized)에서 멈춰 있는지 코드를 역추적하십시오.

- 덤프 정밀 분석:

- 생성된 Thread Dump 텍스트 파일을 복사 또는 다운로드하여 외부 전문 덤프 분석 서비스나 도구를 활용하면 보다 시각적이고 입체적인 스레드 관계도를 분석할 수 있습니다.
- **fastThread.io**: 웹 기반의 간편한 스레드 덤프 정밀 분석 엔진
- **IBM Thread Analyzer (JCA)**: IBM에서 제공하는 전문 오프라인 스레드 덤프 분석 유틸리티

6.10 Heap Status

경로: `/servers/management/heap?agent_id={agent_id}`

Tomcat JVM의 Heap·Non-Heap 메모리 사용량을 **기간별 시계열 차트**와 **실시간·영역별 상세 테이블**로 확인하는 페이지입니다. 메모리 누수나 OOM 징후를 추적할 수 있으며, 필요 시 힙 덤프(`.hprof`) 생성·다운로드·삭제도 함께 제공합니다.

6.10.1 화면



6.10.2 1. 화면 구성

영역	설명
표시 기간	Heap·Non-Heap 차트 조회 범위 선택
Heap / Non-Heap 차트	heap max · heap used , non-heap max · non-heap used 시계열
실시간 메모리 테이블	현재 시점 Heap·Non-Heap Used/Max/Free/Committed
Heap 메모리 상세 사용량	G1 Old Gen, Eden, Survivor 등 풀(Pool)별 사용량
덤프 파일 목록	힙 덤프 생성·다운로드·삭제

6.10.3 2. 표시 기간 (Scope)

상단 **표시 기간** 드롭다운으로 차트에 표시할 시간 범위를 선택합니다. 범위를 바꾸면 **선택한 기간의 Heap 사용량** 데이터를 다시 조회하여 차트를 갱신합니다.



옵션	집계 간격
최근 5분	5초
최근 30분	10초
최근 1시간	30초
최근 12시간	1분

옵션	집계 간격
최근 1일	5분
최근 15일	1시간
최근 30일	2시간

선택한 기간에 맞춰 차트 제목이 Heap 사용량 (최근 N) / Non-Heap 사용량 (최근 N) 형식으로 변경됩니다.

6.10.4 3. Heap / Non-Heap 차트

좌측은 **Heap**, 우측은 **Non-Heap** 사용량 차트입니다.

시리즈	설명
heap max / non-heap max	할당 가능한 최대 메모리
heap used / non-heap used	실제 사용 중인 메모리

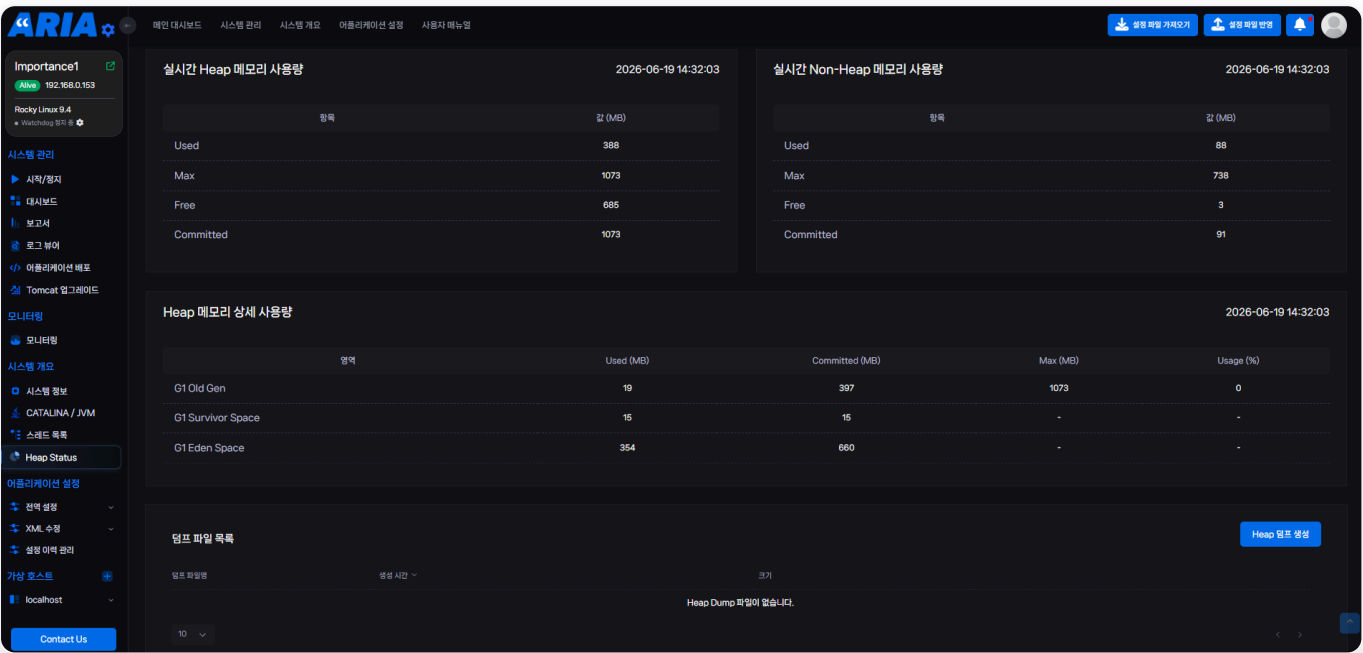
각 차트는 **메인 차트**와 하단 **브러시(미니) 차트**로 구성됩니다. 브러시 영역에서 구간을 드래그해 메인 차트를 확대·이동할 수 있고, 차트 우측 상단 도구로 줌·팬·초기화도 가능합니다.



- **범례 클릭**: 특정 시리즈만 단독 표시하거나, 다시 클릭해 전체 복원
- **Y축 단위**: MB·GB로 자동 변환 표시
- **X축 형식**: 기간에 따라 시·분·일 단위로 자동 조정 (예: 30일 → MMM dd, yyyy)

6.10.5 4. 실시간·상세 메모리 지표

차트 아래 테이블에서 페이지 로드 시점의 스냅샷 값을 확인합니다.



4-1. 실시간 Heap / Non-Heap 메모리

항목	설명
Used	현재 사용 중인 메모리 (MB)
Max	최대 할당 가능 메모리 (MB)
Free	여유 메모리 (MB)
Committed	JVM에 확보된 메모리 (MB)

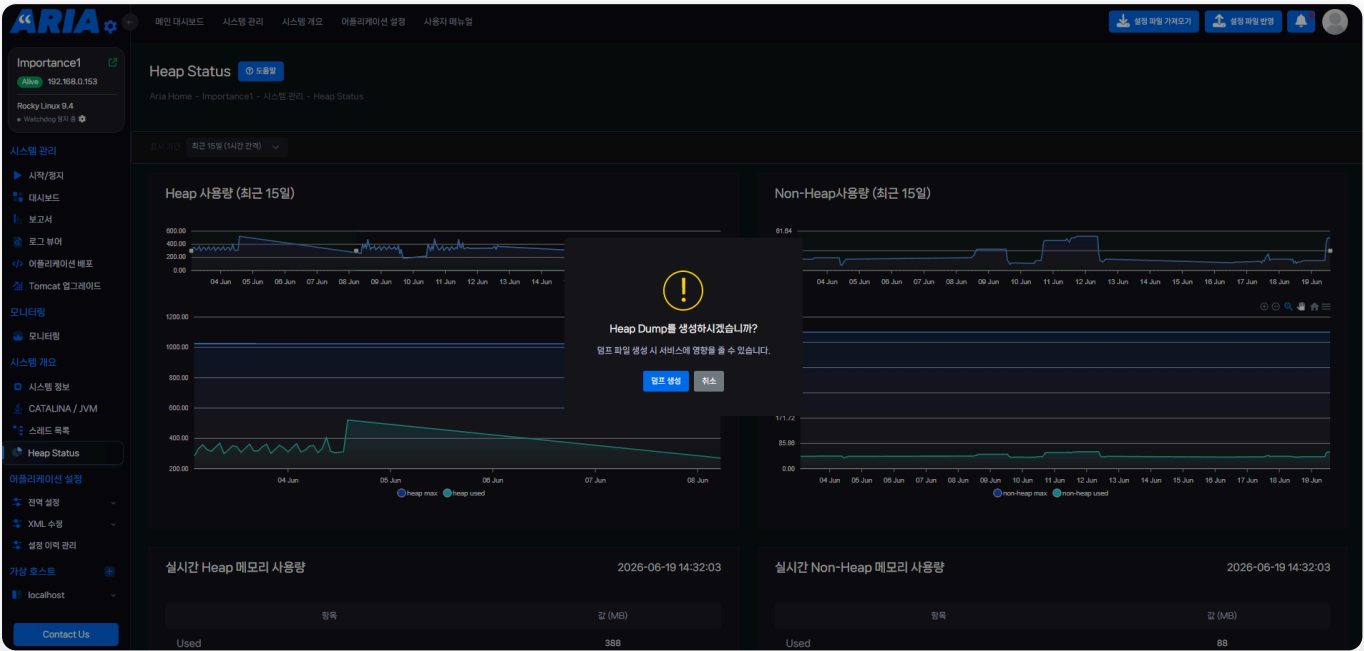
4-2. Heap 메모리 상세 사용량

힙 메모리 풀(영역)별 Used·Committed·Max·Usage(%)를 표시합니다. G1 GC 환경에서는 **G1 Eden Space**, **G1 Survivor Space**, **G1 Old Gen** 등이 노출됩니다.

6.10.6 5. 힙 덤프 생성

Tomcat이 **Alive** 상태일 때만 덤프를 생성할 수 있습니다.

- 1. **Heap 덤프 생성** 버튼을 클릭합니다.
- 2. 확인 모달에서 생성을 확정합니다.



- 1. 서버에서 `jmap` 을 사용해 `.hprof` 파일이 생성됩니다.
- 2. 완료 후 **덤프 파일 목록**에 새 파일이 추가됩니다.

주의: 덤프 생성 중 모든 스레드가 잠시 중지될 수 있어 서비스에 영향을 줄 수 있습니다. `JAVA_HOME` 에 `jmap` 이 없으면 기능이 제한될 수 있습니다. **운영 피크 타이밍을 피해** 실행하세요.

6.10.7 6. 덤프 파일 관리

동작	설명
Dump 다운로드	목록에서 파일을 선택해 로컬 PC로 다운로드 (진행률 모달 표시)
삭제	서버 디스크에서 덤프 파일 영구 삭제

덤프 파일 목록 컬럼

컬럼	설명
덤프 파일명	힙 덤프 파일 이름
생성 시간	덤프가 생성된 시각
크기	파일 용량 (2진 단위 KB/MB/GB)

다운로드한 `.hprof` 파일은 Eclipse MAT, VisualVM, JProfiler 등으로 메모리 누수 분석에 활용할 수 있습니다.

6.10.8 💡 활용 팁

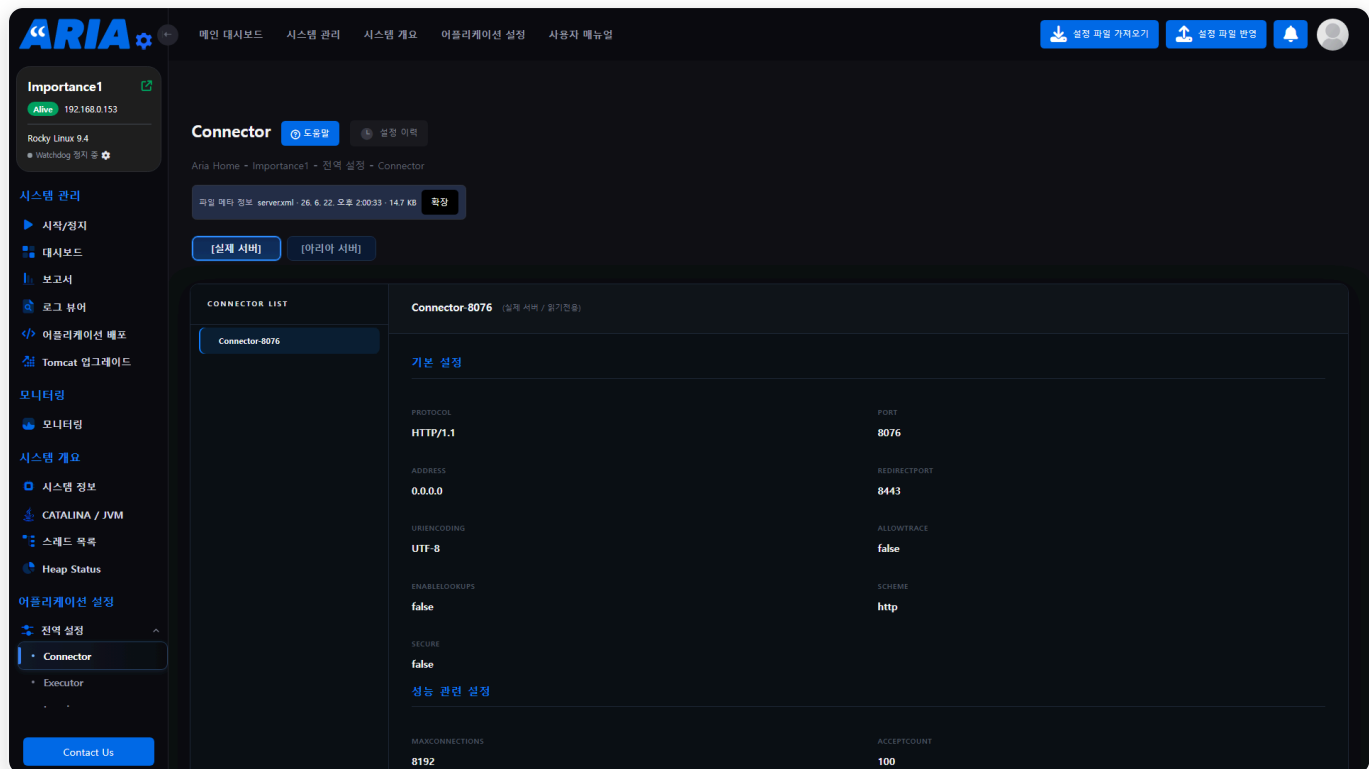
- **메모리 추이 확인:** 표시 기간을 15일·30일로 넓혀 Old Gen 사용량이 GC 이후에도 계속 증가하는지 차트에서 확인하세요.
- **영역별 진단: Heap 메모리 상세 사용량**에서 Eden 대비 Old Gen 점유율이 높아지면 장수 객체 누적을 의심할 수 있습니다.
- **OOM 발생 시:** 오류 직후 힙 덤프를 생성해 재시작 전 메모리 상태를 보존하세요. 덤프 파일 크기는 힙 설정(`-Xmx`)에 가깝게 생성되므로 디스크 여유 공간을 미리 확인하세요.
- **JVM 옵션 튜닝:** 힙 크기·GC 옵션 변경은 [CATALINA / JVM](#) 페이지의 `setenv` 편집기에서 수행합니다.

6.11 전역 설정 - Connector

경로: `/servers/management/config/connector?agent_id={agent_id}`

Tomcat `server.xml` 의 **Connector** 설정을 웹 UI로 편집하는 페이지입니다. HTTP·HTTPS·AJP 등 리스너 포트, 프로토콜, 연결 타임아웃, SSL Host Config를 관리합니다.

6.11.1 화면



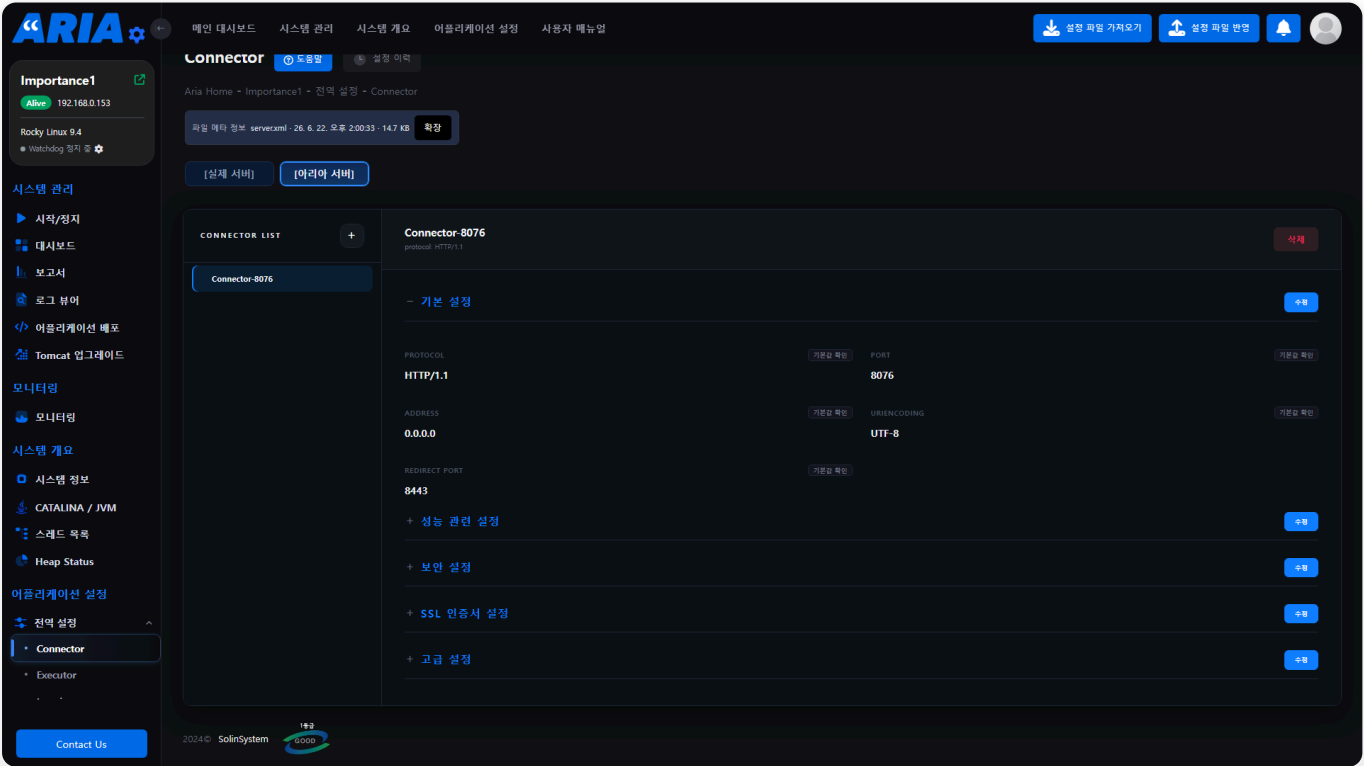
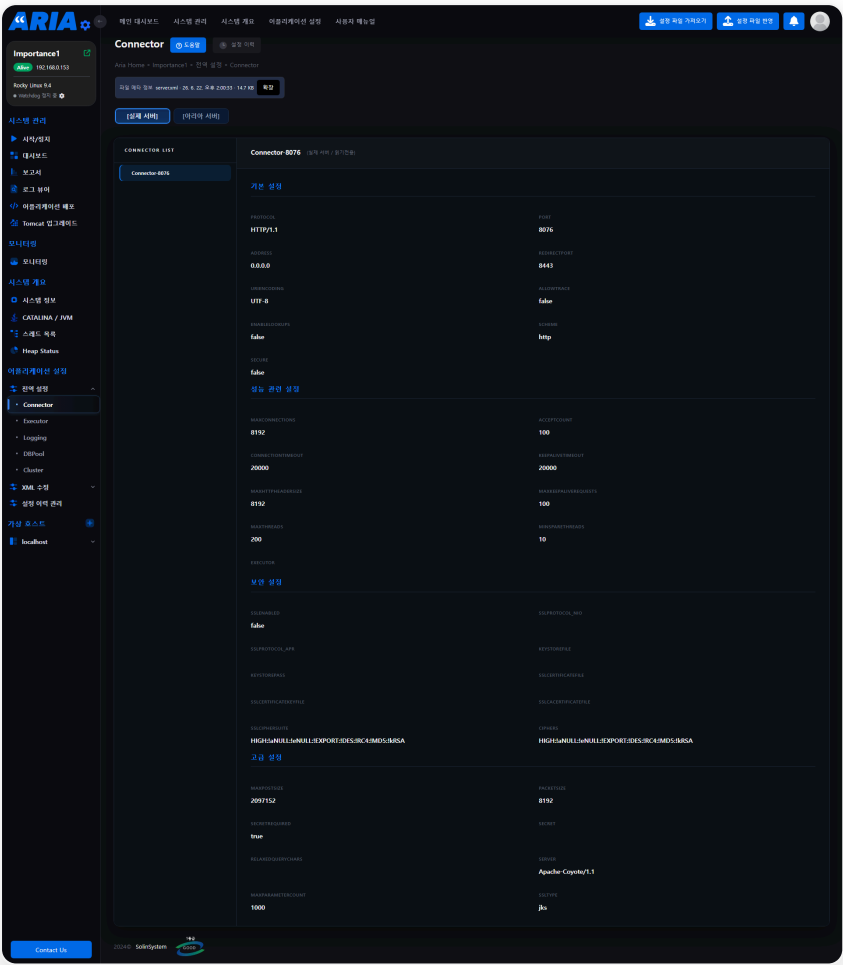
6.11.2 1. 화면 구성

영역	설명
설정 소스 전환	[아리아 서버] / [실제 서버] 탭
설정 파일 메타	아리아 작업본과 실제 <code>server.xml</code> 비교
Connector 목록	좌측 폴더형 목록 (포트별 Connector)
Connector 상세	포트, 프로토콜, <code>connectionTimeout</code> , <code>maxThreads</code> 등
SSL Host Config	HTTPS Connector의 인증서·암호화 스위트 설정
Connector 추가	신규 Connector 등록 모달
수정 / 저장 / 취소	선택 Connector 편집 제어
설정 이력	<code>server.xml</code> 변경 이력 페이지로 이동

2. 설정 소스 전환

[아리아 서버] 탭은 ARIA 작업본의 Connector 설정을 편집하는 화면입니다. **저장** 시 반영 대기 목록에 등록됩니다.

[실제 서버] 탭은 Tomcat에 적용된 Connector 설정을 **읽기 전용**으로 조회합니다.



6.11.3 3. 설정 항목

기본

항목	설명	설정 가이드
Protocol	Connector에서 사용할 프로토콜입니다.	HTTP/1.1, AJP/1.3 등 용도에 맞게 선택
Port	Connector가 수신할 포트 번호입니다.	필수. 다른 프로세스·Connector와 충돌 없는 포트
Address	Connector가 바인딩할 IP 주소입니다. 비우면 모든 인터페이스에서 수신합니다.	특정 NIC만 수신할 때 IP 지정
URIEncoding	URI를 디코딩할 때 사용할 문자 인코딩입니다.	기본 UTF-8 권장
Redirect Port	HTTP 요청을 HTTPS로 리다이렉트할 때 사용할 포트입니다.	HTTP Connector에서 HTTPS 포트 지정

연결·스레드

항목	설명	설정 가이드
maxPostSize	POST 요청 본문의 최대 크기(bytes)입니다.	기본 2097152 (2MB). 대용량 업로드 시 증가
maxConnections	서버가 동시에 허용하는 최대 연결 수입니다.	기본 8192
acceptCount	모든 스레드가 사용 중일 때 대기열에 들어갈 수 있는 최대 요청 수입니다.	기본 100
connectionTimeout	클라이언트 연결 타임아웃(ms)입니다.	기본 60000 (60초)
keepAliveTimeout	Keep-Alive 연결의 타임아웃(ms)입니다.	HTTP 지속 연결 유지 시간
executor	이 Connector가 사용할 Executor의 이름입니다.	Executor 설정에 등록된 이름
maxHttpRequestSize	HTTP 요청·응답 헤더의 최대 크기(bytes)입니다.	기본 8192 (8KB)
packetSize	AJP 프로토콜 패킷 크기입니다.	AJP Connector 전용

항목	설명	설정 가이드
maxKeepAliveRequests	Keep-Alive 연결 하나당 처리할 최대 요청 수입니다.	-1 은 무제한
maxThreads	Connector가 생성할 요청 처리 스레드의 최대 수입니다.	기본 200. <code>executor</code> 미지정 시 직접 제어
minSpareThreads	항상 실행 중인 최소 스레드 수입니다.	기본 10

AJP·기타

항목	설명	설정 가이드
allowTrace	HTTP TRACE 메서드 허용 여부입니다.	기본 <code>false</code> — 보안상 비활성 권장
enableLookups	클라이언트 IP에 대한 DNS 역조회 여부입니다.	기본 <code>false</code> — 성능상 비활성 권장
server	HTTP 응답 <code>Server</code> 헤더에 표시할 값입니다.	필요 시 커스터마이즈
secretRequired	AJP 연결 시 shared secret 필수 여부입니다.	AJP 보안 연동 시 <code>true</code>
secret	AJP shared secret 값입니다.	<code>secretRequired=true</code> 시 설정
relaxedQueryChars	URL 쿼리 문자열에서 허용할 특수 문자입니다.	<code>[,] , {</code> 등 필요 시 지정

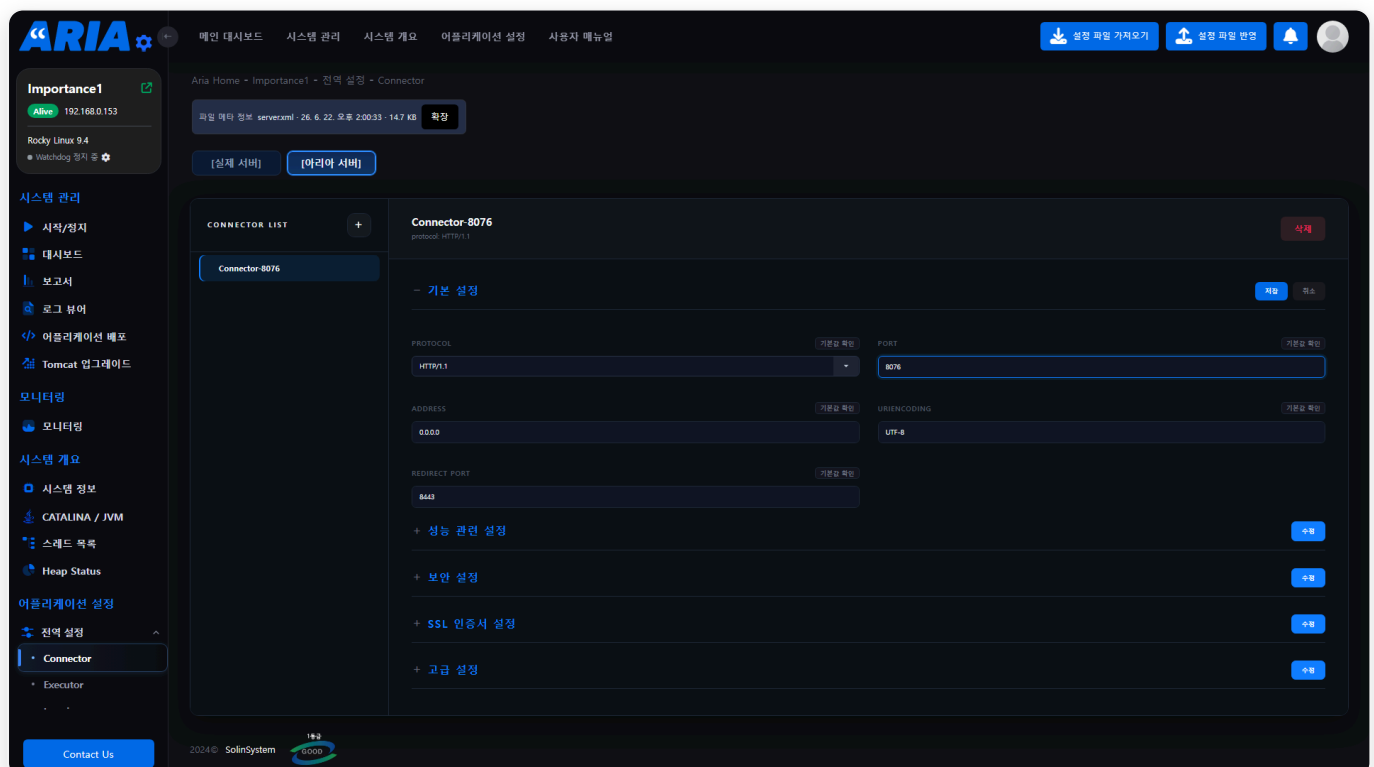
SSL / HTTPS

항목	설명	설정 가이드
SSLEnabled	SSL/TLS 사용 여부입니다.	HTTPS Connector에서 <code>true</code>
secure	secure 스킴 사용 여부입니다.	HTTPS에서 <code>true</code>
scheme	URL 스킴입니다.	HTTPS: <code>https</code>
SSL 설정 방법	인증서를 적용하는 방식(JKS, APR, SNI)입니다.	환경에 맞는 방식 선택
SSLCertificateFile	PEM 형식 서버 인증서 파일 경로(APR)입니다.	APR 방식 시 지정
SSLCACertificateFile	PEM 형식 CA-체인 인증서 파일 경로(APR)입니다.	클라이언트 인증-체인 검증 시

항목	설명	설정 가이드
SSLCertificateKeyFile	PEM 형식 개인키 파일 경로(APR)입니다.	APR 방식 시 지정
SSLCipherSuite	허용할 암호화 스위트 목록입니다.	보안 정책에 맞게 지정
SSLProtocol	사용할 SSL/TLS 프로토콜 버전입니다.	TLSv1.2 이상 권장
keystoreFile	JKS 키스토어 파일 경로입니다.	JKS 방식 시 필수
keystorePass	JKS 키스토어 비밀번호입니다.	JKS 방식 시 필수
ciphers	JKS 방식에서 허용할 암호화 목록입니다.	보안 정책에 맞게 지정
sslProtocol	JKS 방식 SSL 프로토콜입니다.	TLS 권장
hostName	SNI(Server Name Indication) 호스트명입니다.	호스트별 인증서 분기 시 지정

6.11.4 4. Connector 설정 편집

1. [아리아 서버] 탭에서 좌측 목록에서 Connector를 선택합니다.
2. 수정 버튼을 클릭해 편집 모드로 전환합니다.



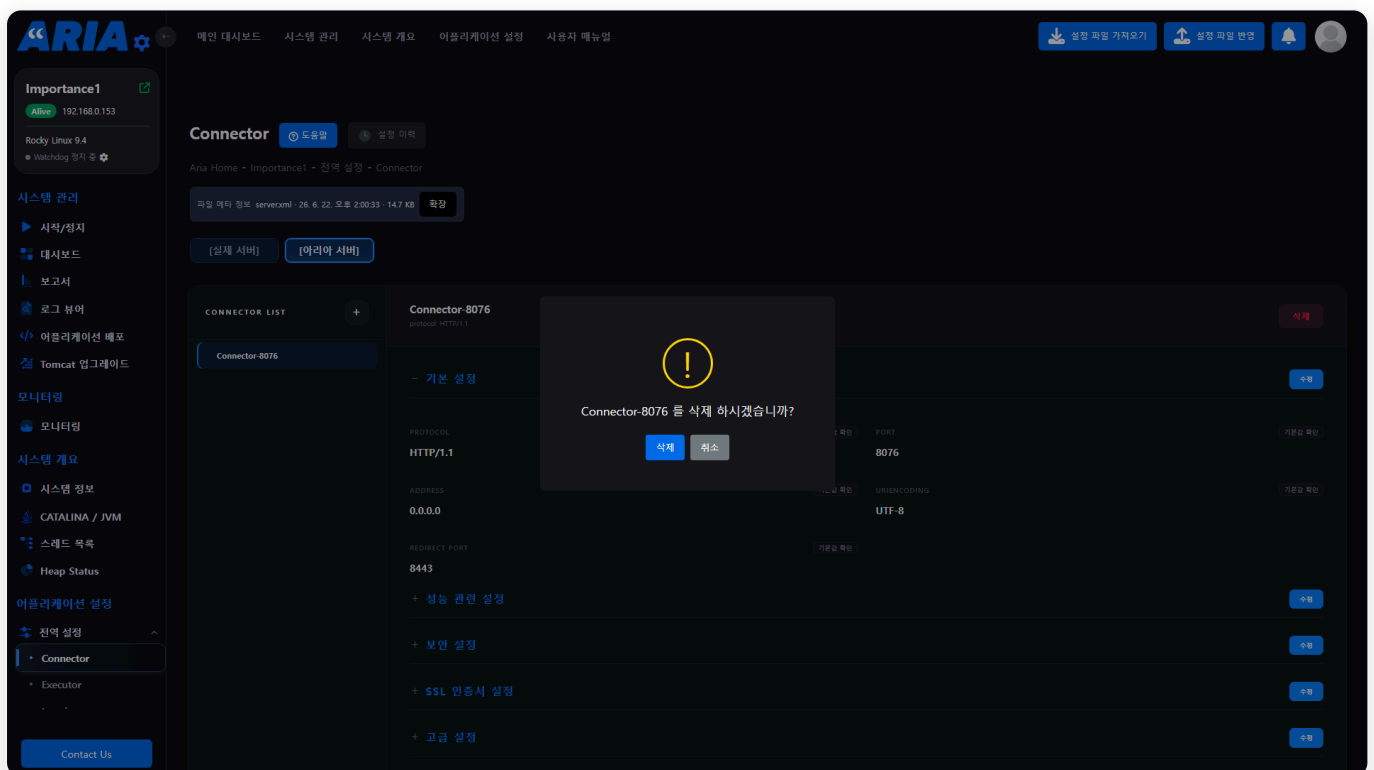
1. **설정 항목**을 참고해 Connector 기본 속성 또는 SSL Host Config를 변경합니다.
2. **저장**을 클릭하면 아리아 작업본에 반영되고 반영 대기 목록에 등록됩니다.

버튼	동작
수정	선택 Connector 편집 모드 진입
저장	아리아 작업본에 저장 (반영 대기)
취소	편집 중 변경 사항 되돌림
Connector 추가	신규 Connector 등록 모달 열기

주의: 저장만으로는 Tomcat에 즉시 적용되지 않습니다.

Connector 삭제

1. 삭제할 Connector를 선택하고 **삭제** 버튼을 클릭합니다.
2. 확인 모달에서 삭제를 확정합니다.



6.11.5 5. 서버 반영

1. 헤더 **반영 대기** 배지를 확인합니다.

2. **설정 파일 반영**에서 대상 서버: `server.xml` 항목을 선택해 반영합니다.

설정 변경 반영 문서를 참고하세요.

6.11.6 💡 활용 팁

- **포트 충돌**: 동일 서버의 다른 Connector 프로세스와 포트가 겹치지 않는지 확인하세요.
- **Executor 연계**: `executor` 속성이 지정된 Connector는 **Executor 설정**과 함께 점검하세요.
- **설정 이력**: **설정 이력** 버튼으로 `server.xml` 변경 내역을 추적할 수 있습니다.

6.12 전역 설정 - Executor

경로: `/servers/management/config/executor?agent_id={agent_id}`

Tomcat `server.xml` 의 **Executor** 스레드 풀 설정을 웹 UI로 편집하는 페이지입니다. Connector가 공유하는 스레드 풀의 `minSpareThreads`, `maxThreads`, `maxQueueSize` 등을 관리합니다.

6.12.1 화면

The screenshot shows the ARIA web interface for configuring the Executor. The main content area displays the 'executor' configuration for the 'Importance1' agent. The configuration is shown in a table with the following values:

PROPERTY	VALUE
MINSPARETHREADS	25
MAXTHREADS	200
MAXIDLETIME	60000
MAXQUEUESIZE	2147483647
THREADRENEWALDELAY	1000
THREADPRIORITY	5
DAEMON	true
NAME	executor
NAMEPREFIX	tomcat-exec-

The left sidebar contains navigation links for System Management, Monitoring, and Application Settings. The top navigation bar includes links for System Overview, System Settings, and Application Settings.

6.12.2 1. 화면 구성

영역	설명
설정 소스 전환	[아리아 서버] / [실제 서버] 탭
설정 파일 메타	아리아 작업본과 실제 <code>server.xml</code> 비교
Executor 목록	등록된 Executor 목록
Executor 상세	<code>namePrefix</code> , <code>maxThreads</code> , <code>maxQueueSize</code> 등
수정 / 저장 / 취소	선택 Executor 편집 제어
설정 이력	<code>server.xml</code> 변경 이력 페이지로 이동

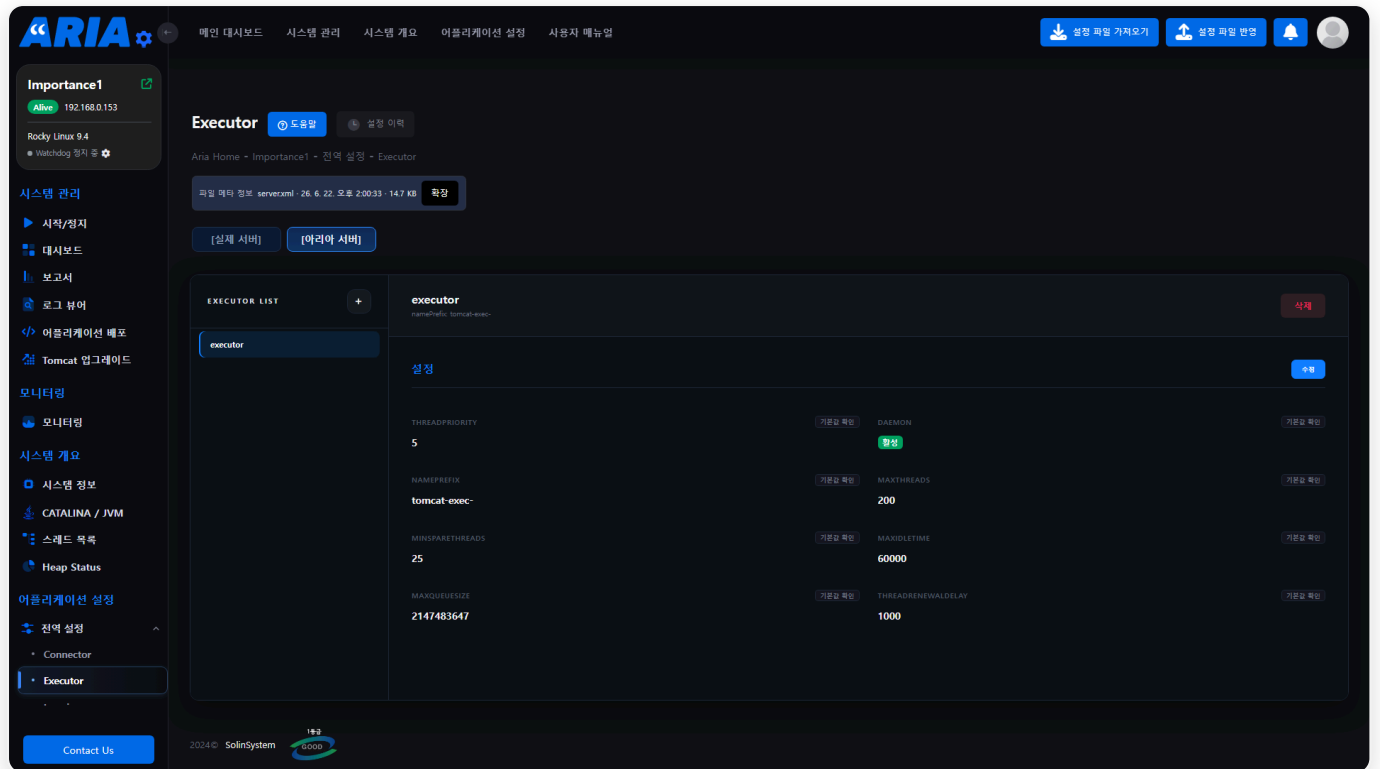
2. 설정 소스 전환

[아리아 서버] 탭은 ARIA 작업본의 Executor 설정을 편집하는 화면입니다. **저장** 시 반영 대기 목록에 등록됩니다.

[실제 서버] 탭은 Tomcat에 적용된 Executor 설정을 읽기 전용으로 조회합니다.

The screenshot displays the ARIA web interface for managing executors. The sidebar on the left provides navigation options, including system management, monitoring, and application settings. The main panel shows the 'Executor' configuration page, which includes a list of executors and a detailed view of the 'executor' configuration. The configuration parameters are as follows:

Parameter	Value
MINSPARETHREADS	25
MAXTHREADS	200
MAXIDLETIME	60000
MAXQUEUESIZE	2147483647
THREADRENEWALDELAY	1000
NAME	executor
THREADPRIORITY	5
NAMEPREFIX	tomcat-exec-
DAEMON	true



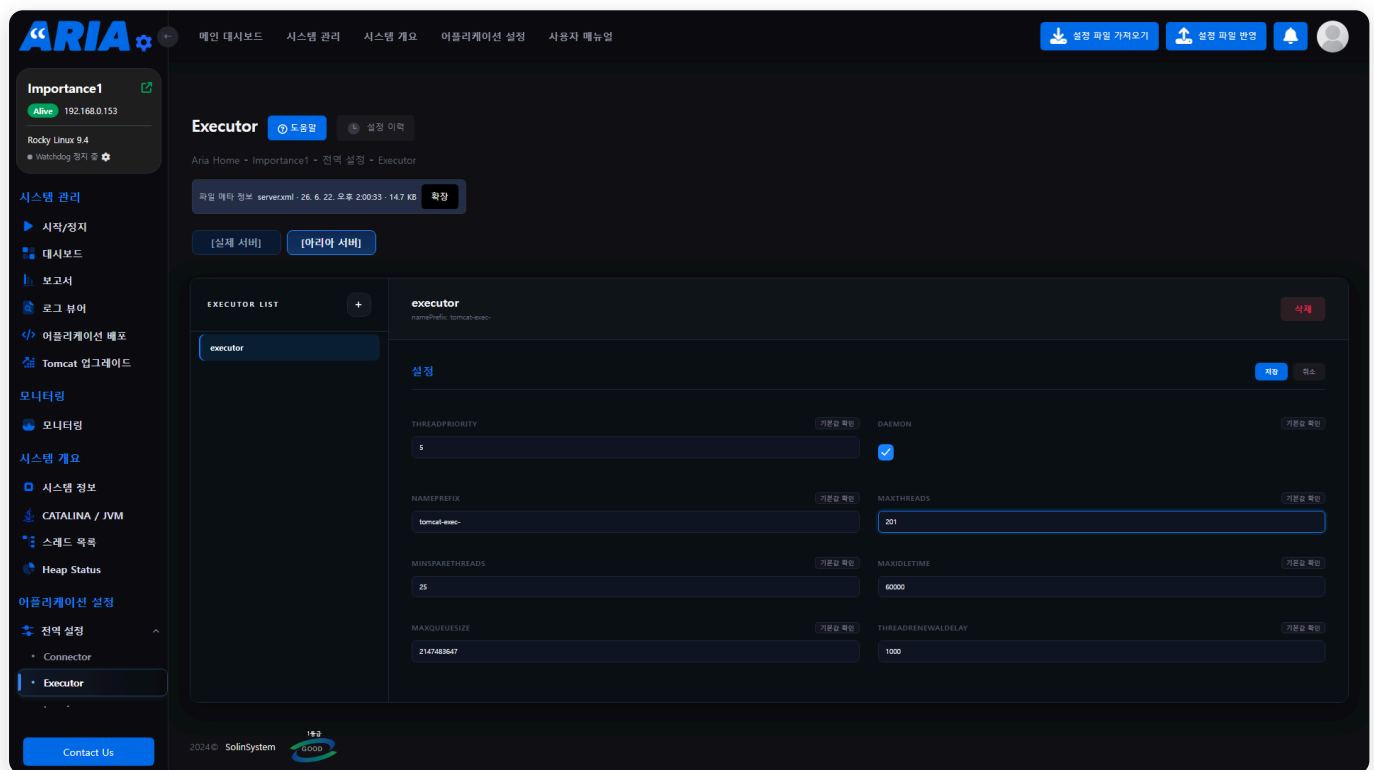
6.12.3 3. 설정 항목

항목	설명	설정 가이드
threadPriority	Executor 안에서 생성되는 스레드의 우선순위입니다.	기본 5 . 일반적으로 변경 불필요
daemon	스레드가 데몬 스레드로 동작할지 여부입니다. 데몬 스레드는 JVM 종료 시 함께 종료됩니다.	기본 true 권장
namePrefix	Executor가 생성하는 스레드 이름의 접두사입니다. 스레드 덤프.로그에서 식별에 사용합니다.	기본 tomcat-exec-
maxThreads	Executor에서 생성할 수 있는 최대 스레드 수입니다.	기본 200 . CPU-메모리에 맞게 조정
minSpareThreads	항상 유지할 최소 유휴 스레드 수입니다.	기본 25
maxIdleTime	유휴 스레드가 종료되기까지의 대기 시간(ms)입니다.	기본 60000
maxQueueSize	스레드가 작업을 거부하기 전 큐에 대기할 수 있는 작업의 최대 수입니다.	기본 무제한. 과부하 방지 시 제한

항목	설명	설정 가이드
threadRenewalDelay	Context 중지 시 Executor 스레드 갱신 지연 시간(ms)입니다. ThreadLocal 누수 방지에 사용됩니다.	기본 1000

6.12.4 4. Executor 설정 편집

1. [아리아 서버] 탭에서 Executor를 선택합니다.
2. 수정 버튼을 클릭해 편집 모드로 전환합니다.

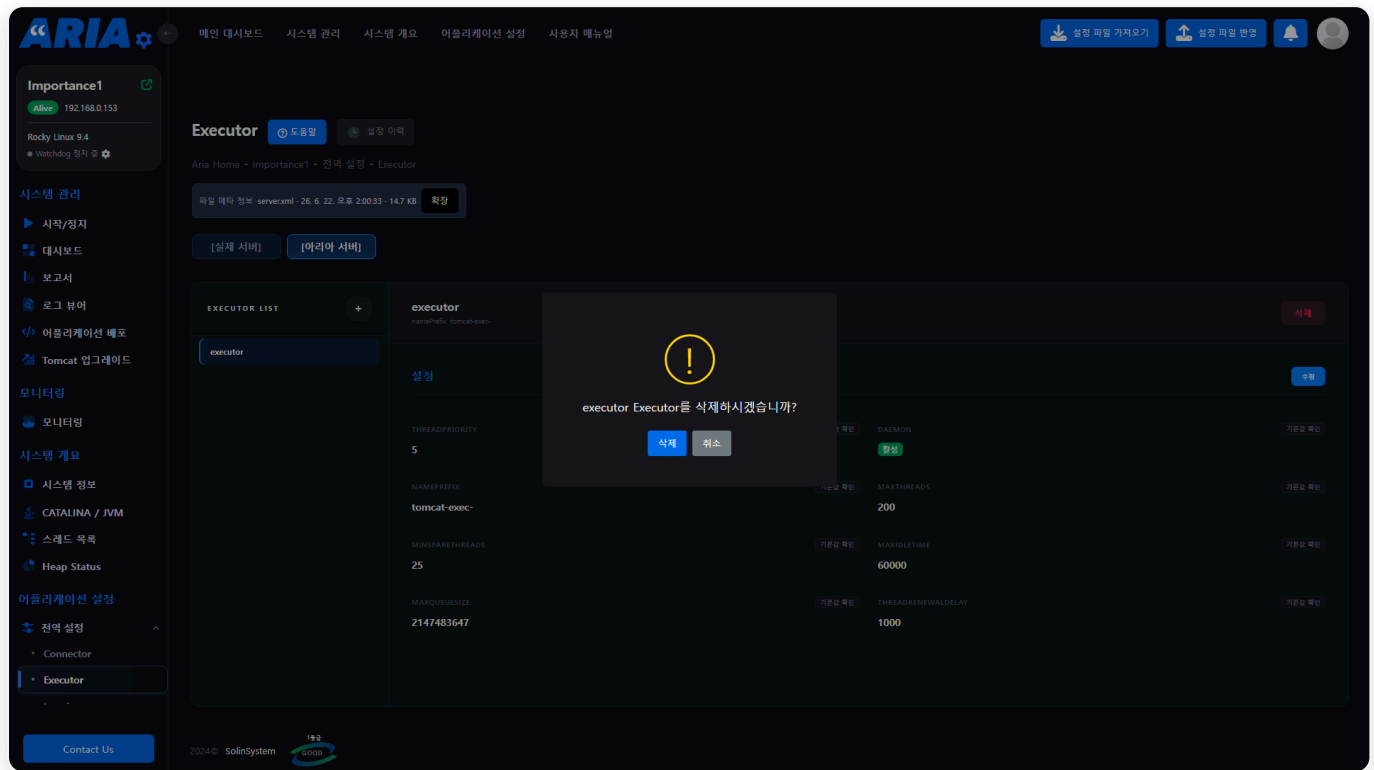


1. **설정 항목**을 참고해 스레드 풀 속성을 변경합니다.
2. **저장**을 클릭하면 아리아 작업본에 반영되고 반영 대기 목록에 등록됩니다.

버튼	동작
수정	선택 Executor 편집 모드 진입
저장	아리아 작업본에 저장 (반영 대기)
취소	편집 중 변경 사항 되돌림

Executor 삭제

1. 삭제할 Executor를 선택하고 **삭제** 버튼을 클릭합니다.
2. 확인 modal에서 삭제를 확정합니다.



6.12.5 5. 서버 반영

설정 변경 반영 절차에 따라 적용하세요.

6.12.6 💡 활용 팁

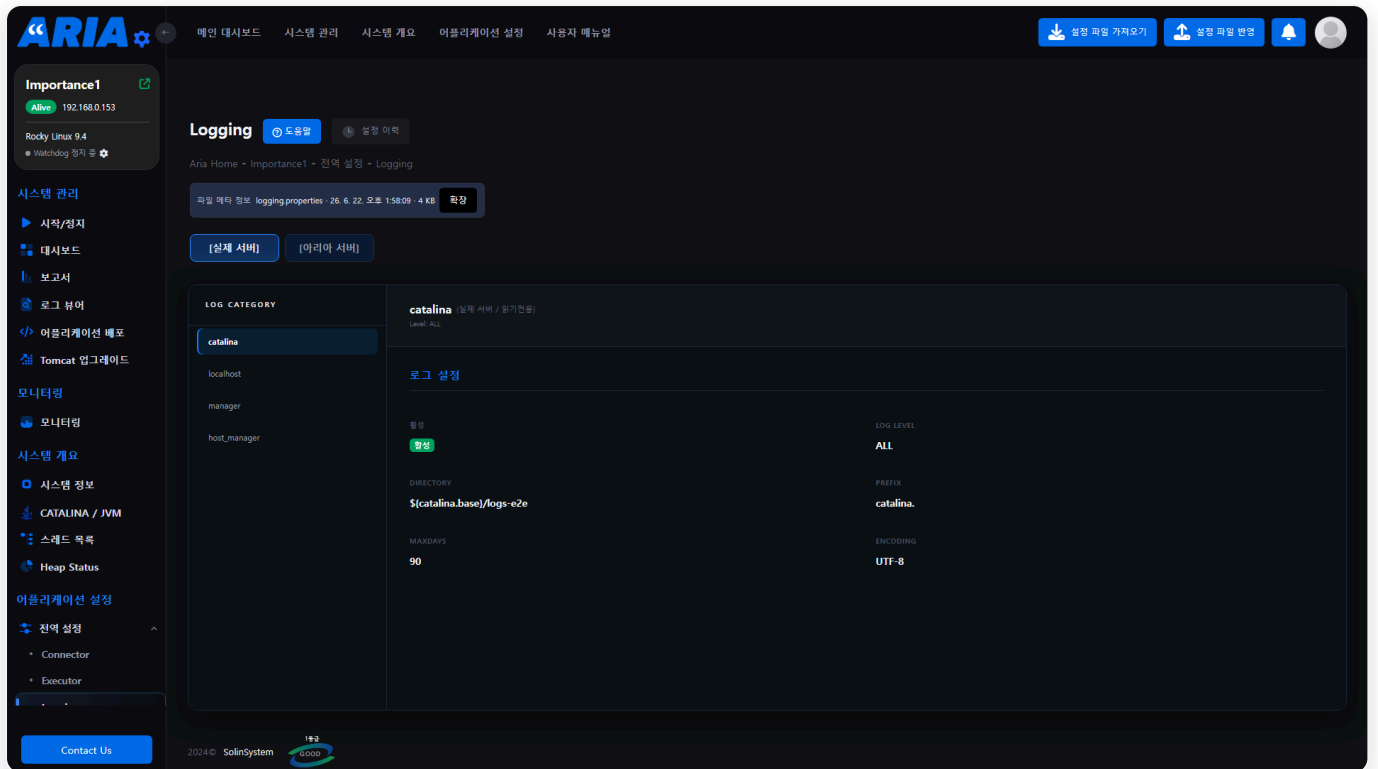
- **Connector 연계:** Executor를 참조하는 Connector의 `maxThreads` 와 함께 점검하세요. ([Connector 설정](#))
- **스레드 모니터링:** [모니터링](#)에서 Tomcat 스레드 사용량을 확인할 수 있습니다.

6.13 전역 설정 - Logging

경로: `/servers/management/config/logging?agent_id={agent_id}`

Tomcat `logging.properties` 의 **로그 카테고리별 Level-Encoding**을 웹 UI로 편집하는 페이지입니다. `catalina`, `localhost`, `manager`, `host-manager` 등 카테고리별 핸들러 설정을 폼으로 관리합니다.

6.13.1 화면



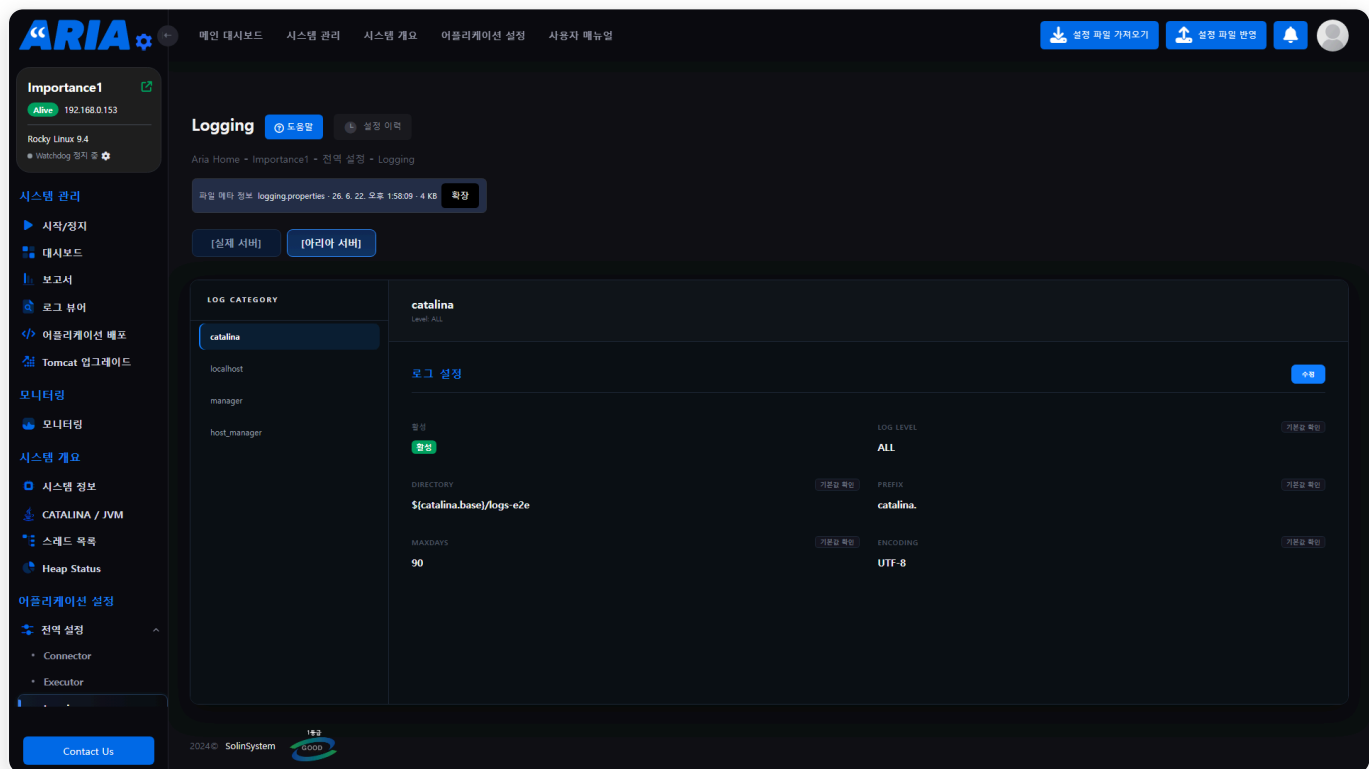
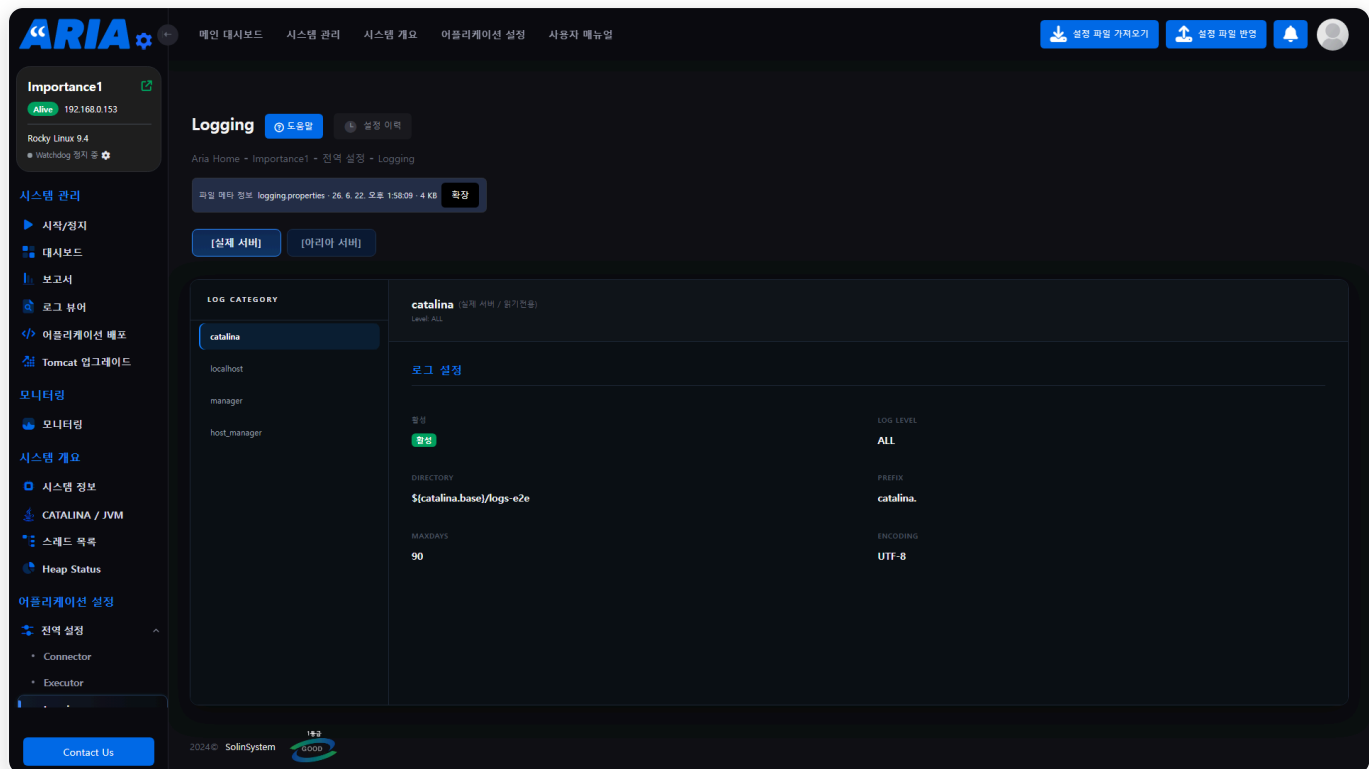
6.13.2 1. 화면 구성

영역	설명
설정 소스 전환	[아리아 서버] / [실제 서버] 탭
설정 파일 메타	아리아 작업본과 실제 logging.properties 비교
LOG CATEGORY 탭	catalina / localhost / manager / host-manager 등
Level / Encoding	카테고리별 로그 레벨·인코딩 설정
수정 / 저장 / 취소	선택 카테고리 편집 제어
설정 이력	logging.properties 변경 이력 페이지로 이동

2. 설정 소스 전환

[아리아 서버] 탭은 ARIA 작업본의 로깅 설정을 편집하는 화면입니다. **저장** 시 반영 대기 목록에 등록됩니다.

[실제 서버] 탭은 Tomcat에 적용된 로깅 설정을 **읽기 전용**으로 조회합니다.



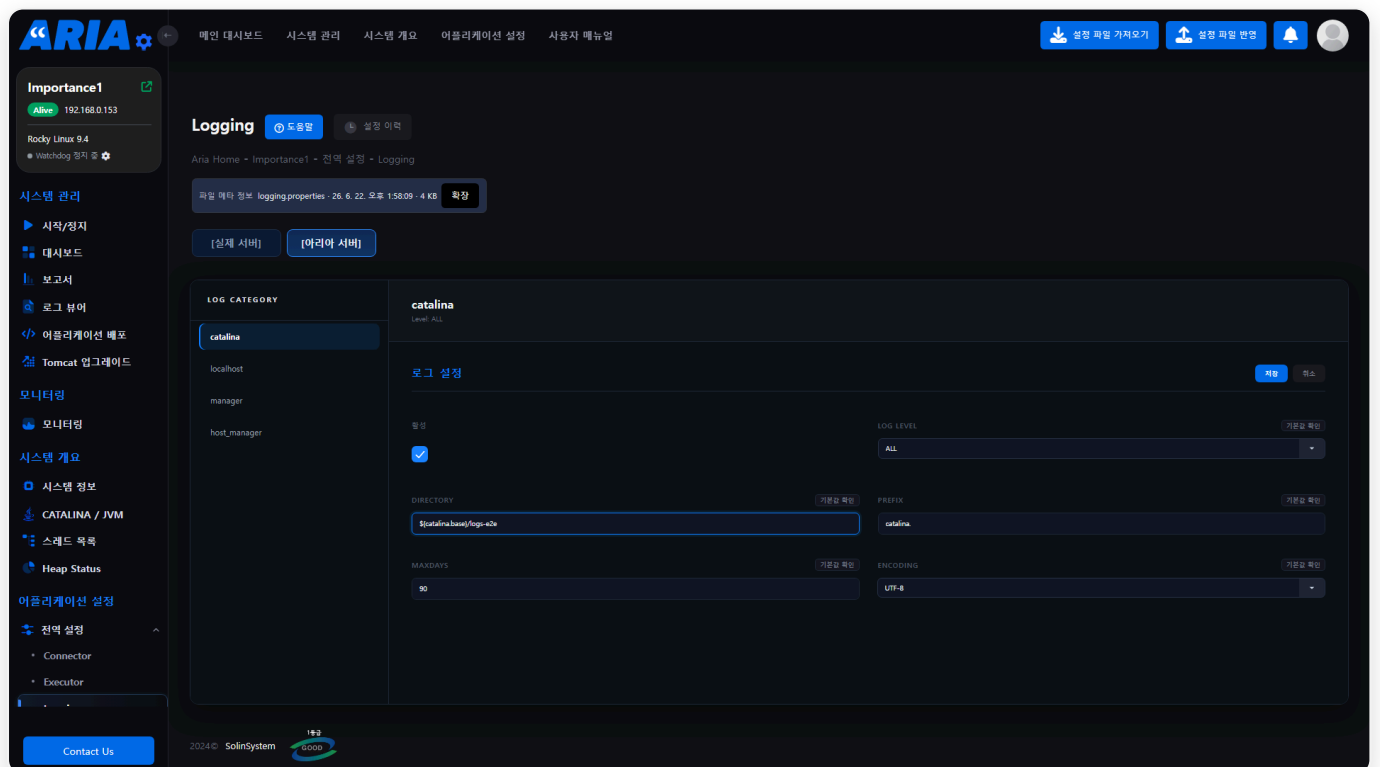
6.13.3 3. 설정 항목

각 LOG CATEGORY(catalina, localhost, manager, host-manager 등)마다 아래 항목이 동일하게 제공됩니다.

항목	설명	설정 가이드
활성	해당 카테고리의 FileHandler를 사용할지 여부입니다. 비활성화하면 해당 카테고리 로그 파일이 생성되지 않습니다.	불필요한 카테고리는 비활성화해 디스크 절약
Log Level	해당 카테고리의 로그 출력 레벨입니다. 지정 레벨 이상의 메시지만 기록됩니다.	운영: INFO ~ WARNING . 장애 분석: FINE 이하
Directory	로그 파일이 저장될 디렉터리입니다. 상대 경로는 <code>\${catalina.base}</code> 기준입니다.	예: logs . 쓰기 권한 확인
Prefix	로그 파일 이름 앞에 붙는 접두사입니다.	예: catalina. → catalina.2026-06-22.log
MaxDays	보관할 로그 파일의 최대 일수입니다. 초과분은 자동 삭제됩니다.	디스크 용량에 맞게 설정
Encoding	로그 파일에 사용할 문자 인코딩입니다.	UTF-8 권장. OS-로그 뷰어와 일치시킬 것

6.13.4 4. Logging 설정 편집

1. [아리아 서버] 탭에서 LOG CATEGORY 탭을 선택합니다.
2. 수정 버튼을 클릭해 편집 모드로 전환합니다.



1. [설정 항목](#)을 참고해 Level·Encoding 등을 변경합니다.
2. **저장**을 클릭하면 아리아 작업본에 반영되고 반영 대기 목록에 등록됩니다.

버튼	동작
수정	선택 카테고리 편집 모드 진입
저장	아리아 작업본에 저장 (반영 대기)
취소	편집 중 변경 사항 되돌림

주의: 저장만으로 Tomcat 로그 설정이 즉시 적용되지 않습니다.

6.13.5 5. 서버 반영

[설정 변경 반영](#) 절차에 따라 `logging.properties` 를 Tomcat에 적용하세요.

6.13.6 💡 활용 팁

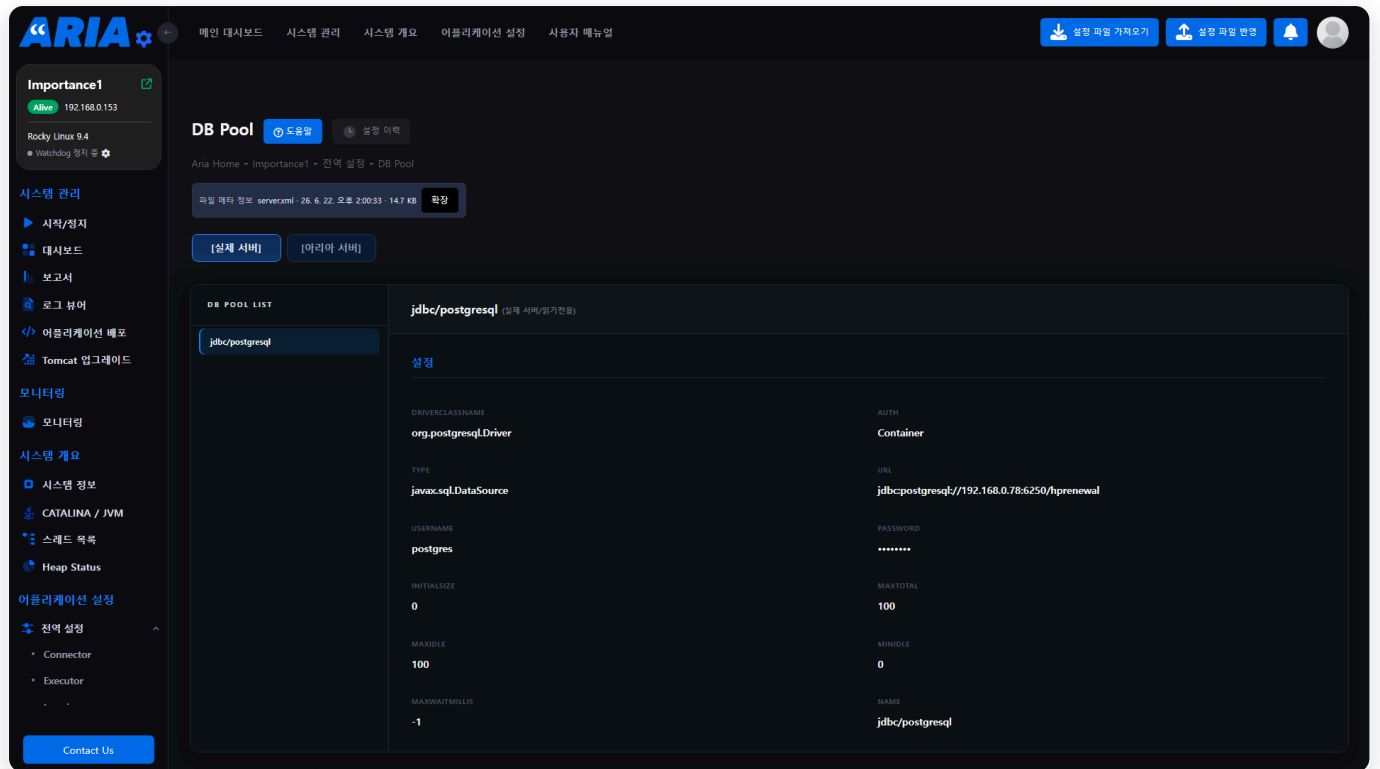
- **원시 파일 편집:** 핸들러·Appender를 직접 수정하려면 [logging.properties](#) 페이지를 사용하세요.
- **로그 확인:** 반영 후 [로그 뷰어](#)에서 실제 로그 출력을 확인하세요.

6.14 전역 설정 - DBPool

경로: `/servers/management/config/dbpool?agent_id={agent_id}`

Tomcat `server.xml` 의 **GlobalNamingResources / JDBC Resource(DB Pool)** 설정을 웹 UI로 편집하는 페이지입니다.

6.14.1 화면



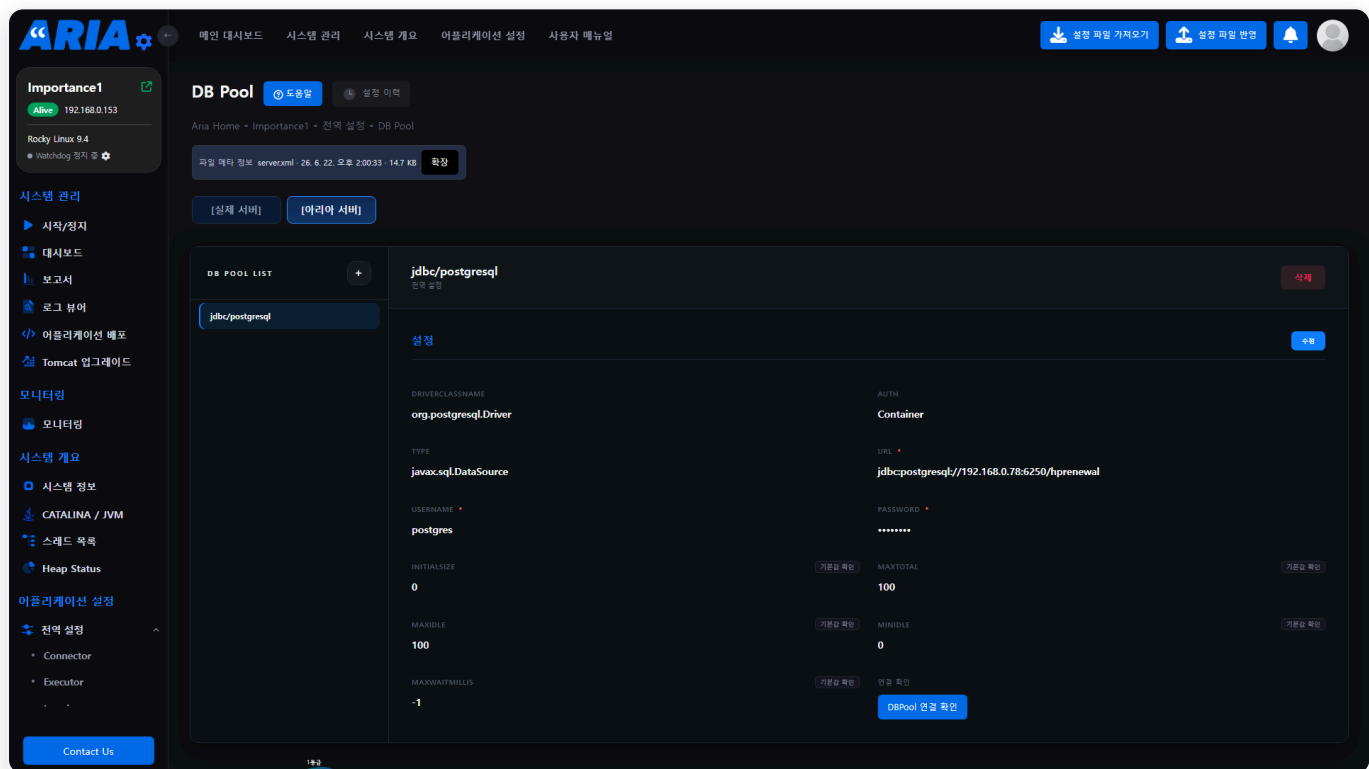
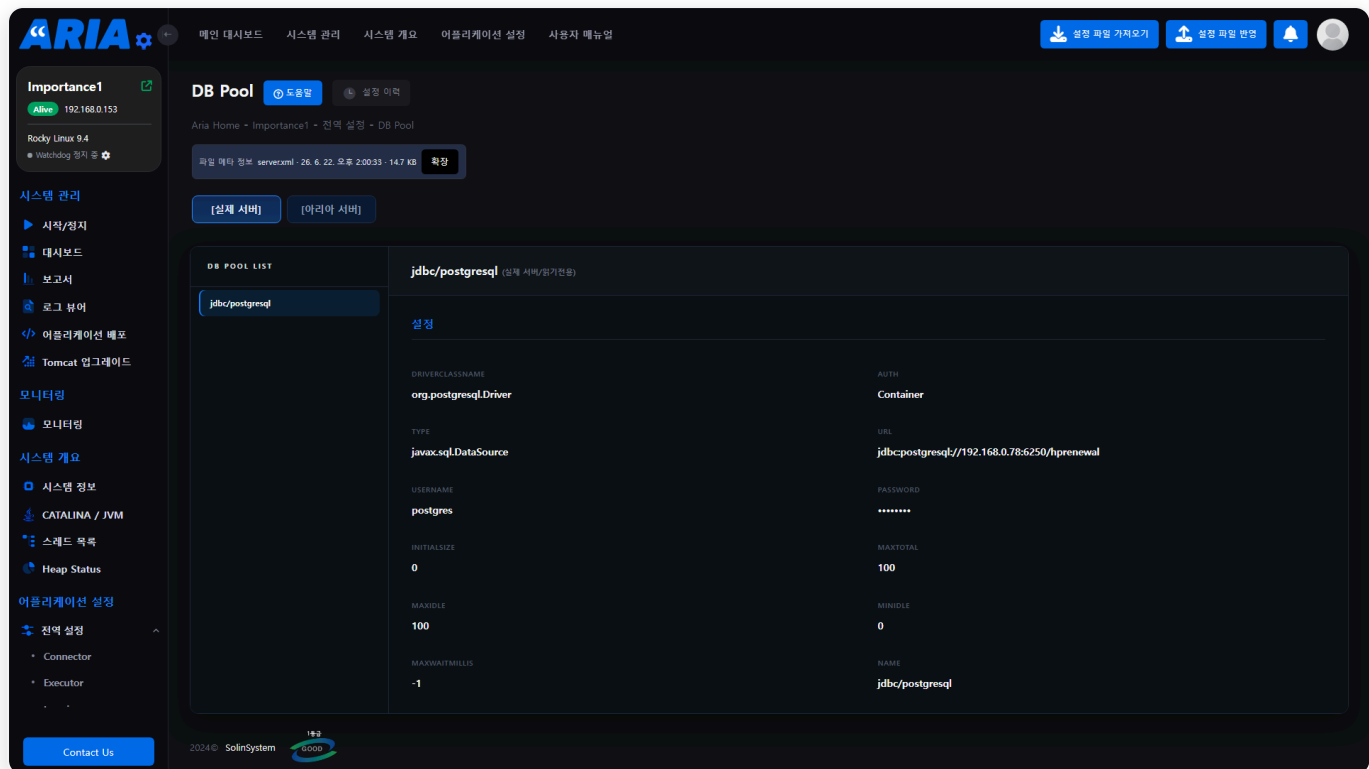
6.14.2 1. 화면 구성

영역	설명
설정 소스 전환	[아리아 서버] / [실제 서버] 탭
설정 파일 메타	아리아 작업본과 실제 <code>server.xml</code> 비교
DB Pool 목록	등록된 JDBC Resource 목록
JDBC Resource 폼	url, 드라이버, username, 풀 크기 등
DBPool 연결 확인	저장 전 DB 연결 테스트
수정 / 저장 / 취소	선택 풀 편집 제어
설정 이력	<code>server.xml</code> 변경 이력 페이지로 이동

2. 설정 소스 전환

[아리아 서버] 탭은 ARIA 작업본의 JDBC Resource를 편집하는 화면입니다. **저장** 시 반영 대기 목록에 등록됩니다.

[실제 서버] 탭은 Tomcat에 적용된 DB Pool 설정을 **읽기 전용**으로 조회합니다.



6.14.3 3. 설정 항목

JDBC 연결

항목	설명	설정 가이드
driverClassName	JDBC 드라이버의 클래스 이름입니다.	DBMS에 맞는 드라이버 지정 (예: <code>org.mariadb.jdbc.Driver</code>)
auth	Connection 인증 방식입니다. Tomcat 컨테이너가 계정을 관리합니다.	<code>Container</code> 고정
type	JNDI Resource 타입입니다.	<code>javax.sql.DataSource</code> 고정
url	Database 연결에 필요한 JDBC URL입니다.	필수. 호스트·포트·DB명 포함 (예: <code>jdbc:mariadb://host:3306/db</code>)
username	Database 연결에 필요한 사용자 이름입니다.	필수
password	Database 연결에 필요한 비밀번호입니다.	필수. 반영 후에도 안전하게 관리

커넥션 풀

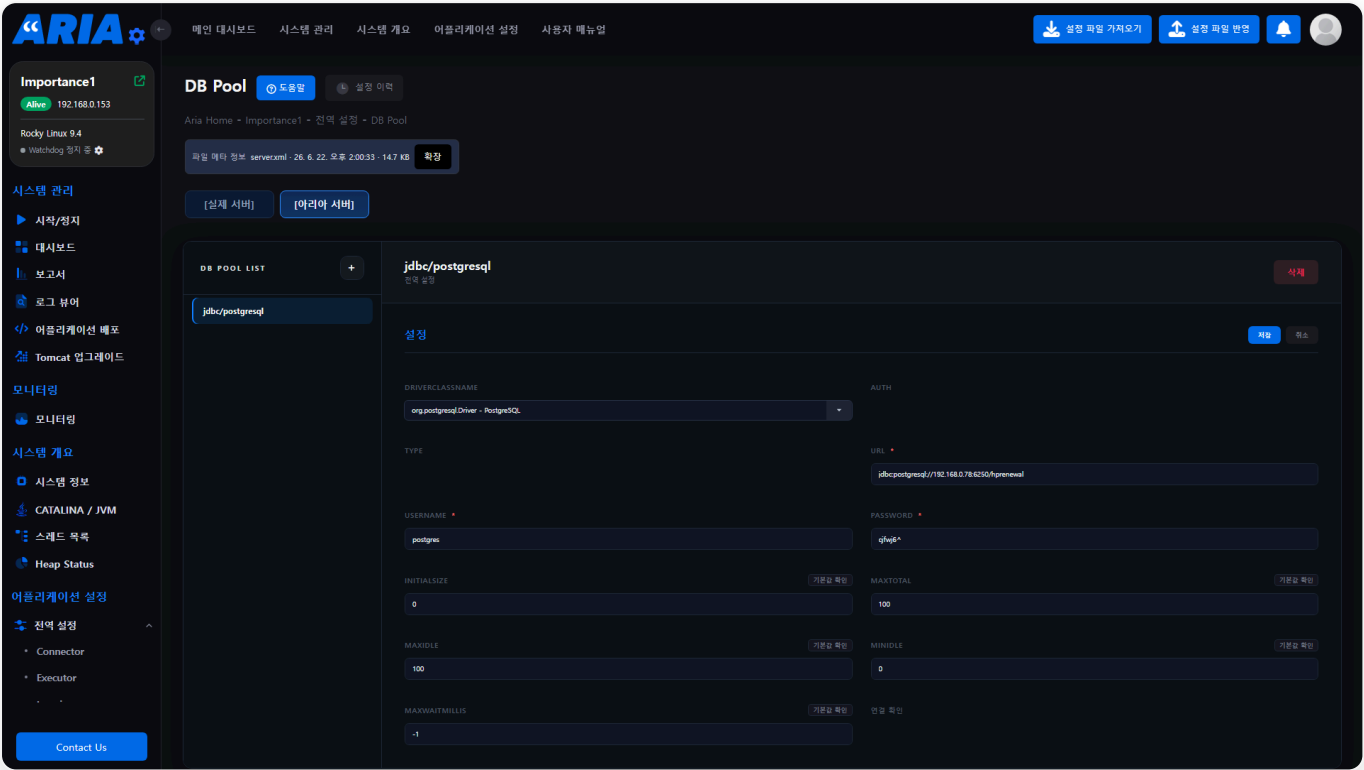
항목	설명	설정 가이드
initialSize	DB Pool 최초 구성 시 생성될 Connection 수입니다.	기본 0. 기동 시 미리 확보할 연결 수
maxTotal	DB Pool에서 사용 가능한 최대 Connection 수입니다. 음수는 무제한입니다.	기본 8. DB 서버 <code>max_connections</code> 이하로 설정
maxIdle	DB Pool에서 유지할 최대 유휴 Connection 수입니다. 음수는 무제한입니다.	기본 8. <code>maxTotal</code> 이하 권장
minIdle	최소 유휴 Connection 수입니다.	트래픽에 따라 0 ~ <code>initialSize</code> 수준
maxWaitMillis	사용 가능한 Connection이 없을 때 대기하는 최대 시간(ms)입니다. -1 은 무한 대기입니다.	기본 -1. 풀 고갈 시 타임아웃이 필요하면 양수 지정

연결 확인

항목	설명	설정 가이드
연결 확인	저장 전 JDBC URL·계정으로 실제 DB 접속을 테스트하는 기능입니다.	저장·반영 전 반드시 실행 권장

6.14.4 4. DB Pool 설정 편집

- 1. [아리아 서버] 탭에서 편집할 JDBC Resource를 선택합니다.
- 2. 수정 버튼을 클릭해 편집 모드로 전환합니다.

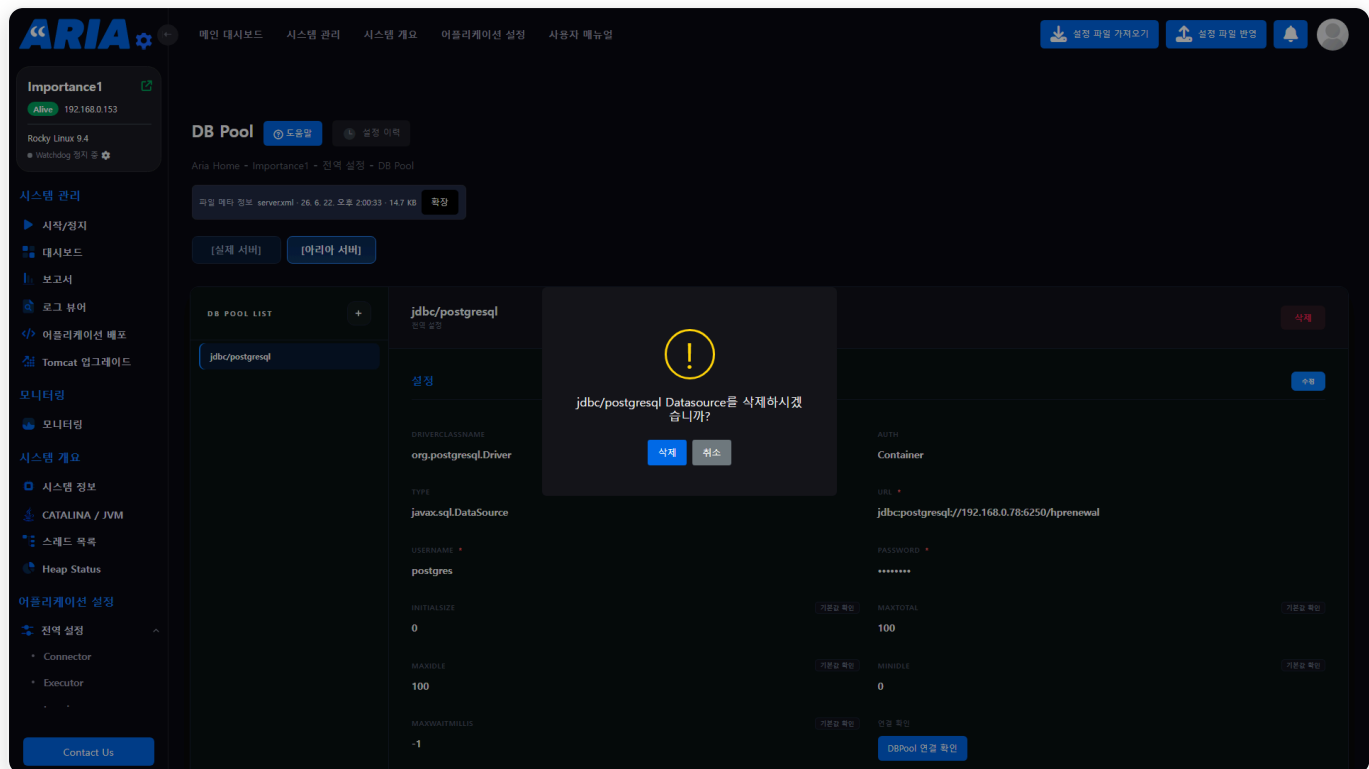


- 1. 설정 항목을 참고해 URL·계정·풀 크기 등을 변경합니다.
- 2. (권장) DBPool 연결 확인으로 접속을 테스트합니다.
- 3. 저장을 클릭하면 아리아 작업본에 반영되고 반영 대기 목록에 등록됩니다.

버튼	동작
수정	선택 풀 편집 모드 진입
DBPool 연결 확인	JDBC URL·계정으로 연결 테스트
저장	아리아 작업본에 저장 (반영 대기)
취소	편집 중 변경 사항 되돌림

DB Pool 삭제

- 1. 삭제할 JDBC Resource를 선택하고 삭제 버튼을 클릭합니다.
- 2. 확인 모달에서 삭제를 확정합니다.



6.14.5 5. 서버 반영

설정 변경 반영 절차에 따라 `server.xml` 변경 사항을 Tomcat에 적용하세요.

6.14.6 💡 활용 팁

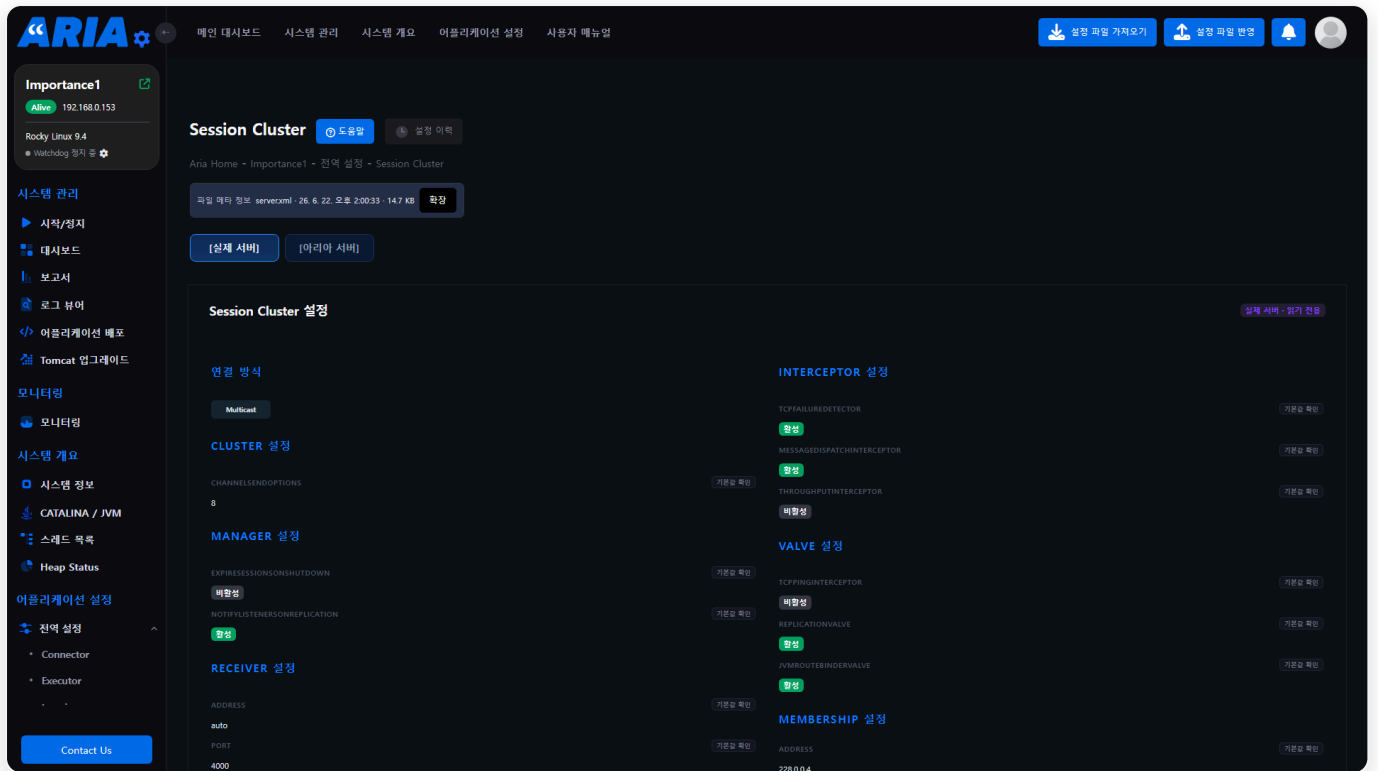
- **연결 확인:** 반영 전 **DBPool 연결 확인**으로 URL·계정 오류를 미리 발견하세요.
- **모니터링:** 반영 후 **모니터링**에서 DataSource Active/Idle 연결 수를 확인할 수 있습니다.
- **가상 호스트별 풀:** vhost 범위 풀은 [가상 호스트 - DBPool](#)을 참고하세요.

6.15 전역 설정 - Cluster

경로: `/servers/management/config/cluster?agent_id={agent_id}`

Tomcat `server.xml` 의 **Session Cluster** 설정을 웹 UI로 편집하는 페이지입니다. 클러스터 Manager·Channel·Receiver와 Static/Multicast 멤버를 구성하고, 변경 내용을 아리아 작업본에 저장한 뒤 서버에 반영할 수 있습니다.

6.15.1 화면



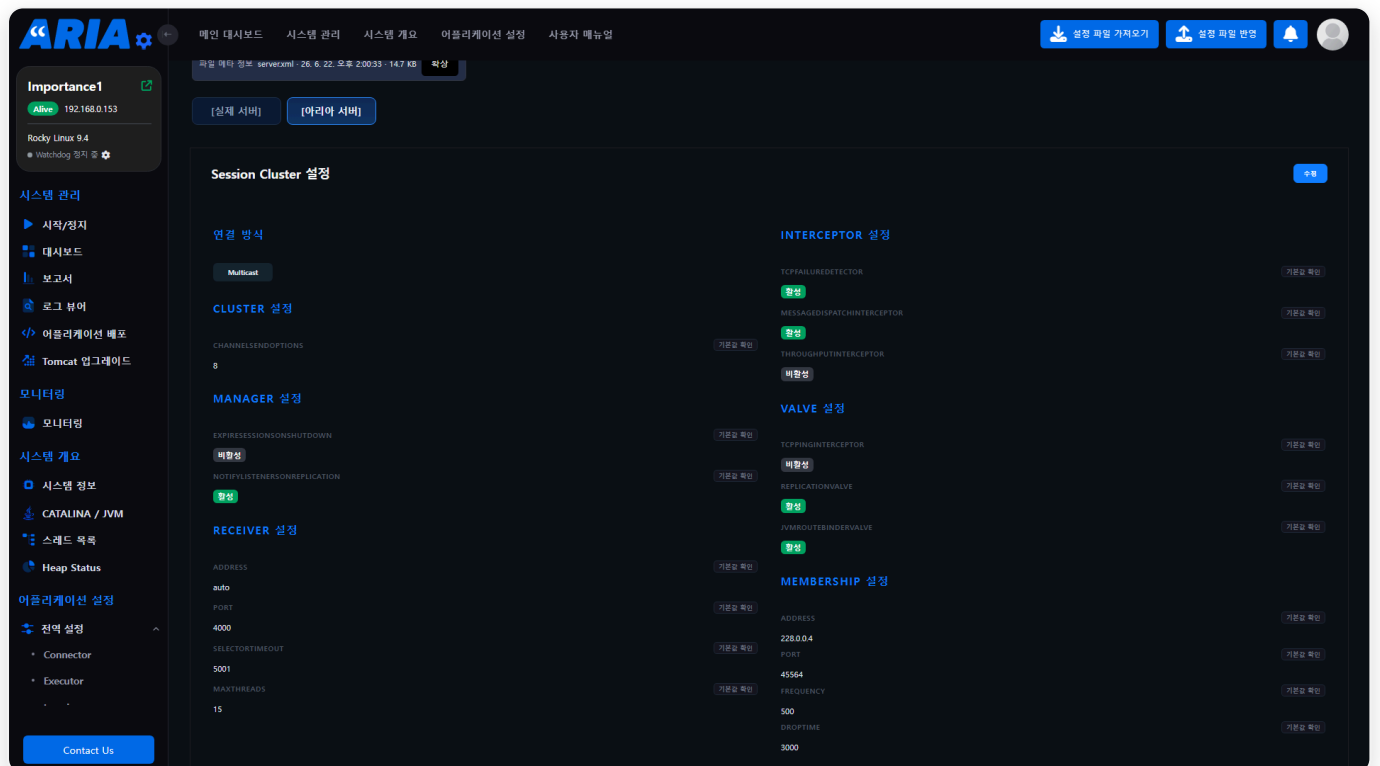
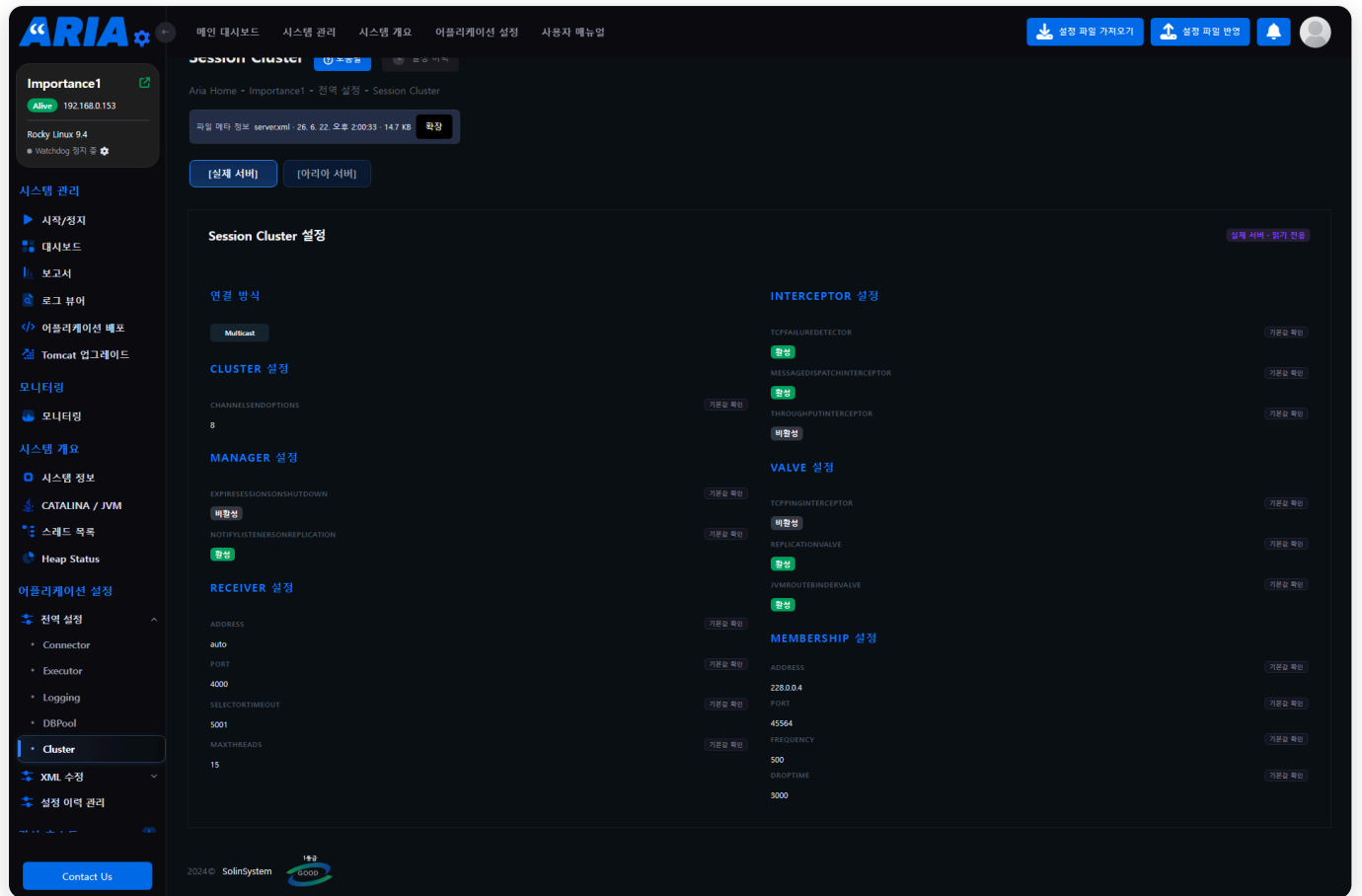
6.15.2 1. 화면 구성

영역	설명
설정 소스 전환	[아리아 서버] / [실제 서버] 탭
설정 파일 메타	아리아 작업본과 실제 Tomcat <code>server.xml</code> 의 수정 시각·크기·반영 상태 비교
Cluster 설정 품	Manager, Channel, Receiver 속성
Static / Multicast 멤버	클러스터 멤버 목록 추가·삭제
수정 / 저장 / 취소	선택 클러스터 항목 편집 제어
설정 이력	<code>server.xml</code> 변경 이력 페이지로 이동

2. 설정 소스 전환

[아리아 서버] 탭은 ARIA 작업본을 편집하는 화면입니다. 기본으로 열리며, **저장** 시 반영 대기 목록에 등록됩니다.

[실제 서버] 탭은 Tomcat에 현재 적용된 Session Cluster 설정을 **읽기 전용**으로 조회합니다. Agent·Manager가 모두 중지된 경우 비활성화될 수 있습니다.



상단 **설정 파일 메타** 패널에서 아리아 작업본과 실제 서버 파일의 최종 수정 시각·크기 차이, **미반영** 여부를 확인할 수 있습니다.
원격 Tomcat 설정이 로컬 작업본과 달라진 경우, 진입 시 최신 설정을 다시 불러올지 묻는 안내가 표시될 수 있습니다.

6.15.3 3. 설정 항목

연결 방식

항목	설명	설정 가이드
Multicast	멀티캐스트 주소를 통해 클러스터 노드 간 메시지를 전송하는 방식입니다. 노드가 동적으로 추가·제거될 때 유용합니다.	MEMBERSHIP 구역(<code>address</code> , <code>port</code> 등)을 설정합니다. Receiver <code>address</code> 에는 <code>auto</code> 또는 유효 IP를 입력합니다.
Static	각 노드의 IP·포트를 명시적으로 지정해 통신하는 방식입니다. 노드 구성이 고정된 환경에 적합합니다.	Static 멤버 목록에 <code>host</code> , <code>port</code> 를 등록합니다. Receiver <code>address</code> 에는 고정 IP를 입력합니다.

Cluster

항목	설명	설정 가이드
<code>channelSendOptions</code>	클러스터 내 데이터 전송 옵션입니다. 전송 신뢰성·복제 횟수 등을 조절해 무결성과 전송 속도를 맞출 수 있습니다.	기본값 8 . Tomcat Cluster 문서와 환경에 맞게 조정합니다.

Manager

항목	설명	설정 가이드
<code>expireSessionsOnShutdown</code>	서버가 정상 종료될 때 세션을 만료시킬지 여부입니다. 활성화하면 종료 시 세션 데이터 손실을 방지합니다.	<code>true</code> 권장
<code>notifyListenersOnReplication</code>	세션 복제 시 세션 리스너에게 이벤트를 알릴지 여부입니다. 세션 상태 변경 후처리가 필요할 때 사용합니다.	<code>true</code> 권장

Receiver

항목	설명	설정 가이드
<code>address</code>	클러스터 메시지를 수신할 IP 주소입니다. 지정한 주소를 기준으로 노드 간 통신을 관리합니다.	Multicast: <code>auto</code> 또는 유효 IP. Static: 고정 IP (예: <code>192.168.0.10</code>)
<code>port</code>	클러스터 메시지를 수신할 포트입니다. 이 포트로 노드 간 메시지를 주고받습니다.	기본 <code>4000</code> . 방화벽에서 노드 간 통신을 허용해야 합니다.

항목	설명	설정 가이드
selectorTimeout	메시지 수신 선택터의 타임아웃(ms)입니다. 지정 시간 내 메시지를 받지 못하면 타임아웃 처리합니다.	기본 5000. 숫자만 입력
maxThreads	클러스터 메시지를 병렬 처리할 최대 스레드 수입니다.	기본 6. 트래픽에 따라 조정

Interceptor

항목	설명	설정 가이드
TcpFailureDetector	TCP 연결로 다른 노드의 가용성을 확인하고 네트워크 실패를 감지하는 인터셉터입니다.	클러스터 안정성 확보 시 활성화 권장
MessageDispatchInterceptor	클러스터 노드 간 메시지 전달을 관리하고 전송 중 문제를 처리하는 인터셉터입니다.	일반적으로 활성화
ThroughputInterceptor	클러스터 통신 처리량을 추적·통계 제공하는 인터셉터입니다.	성능 분석이 필요할 때 활성화
TcpPingInterceptor	주기적으로 다른 노드에 TCP 핑을 보내 노드 상태를 확인하는 인터셉터입니다.	장애 조기 감지가 필요할 때 활성화

Valve

항목	설명	설정 가이드
ReplicationValve	요청 처리 시 세션 데이터 변경을 다른 클러스터 노드에 복제하는 밸브입니다.	세션 복제가 필요하면 활성화 (필수에 가까움)
JvmRouteBinderValve	로드밸런서와 협력해 세션을 특정 JVM 노드에 바인딩하는 밸브입니다.	스티키 세션·로드밸런서 연동 시 활성화

Membership (Multicast)

항목	설명	설정 가이드
address	멀티캐스트 주소입니다. 클러스터 노드가 서로를 발견할 때 사용합니다.	기본 228.0.0.4. 네트워크에서 허용된 멀티캐스트 대역 사용

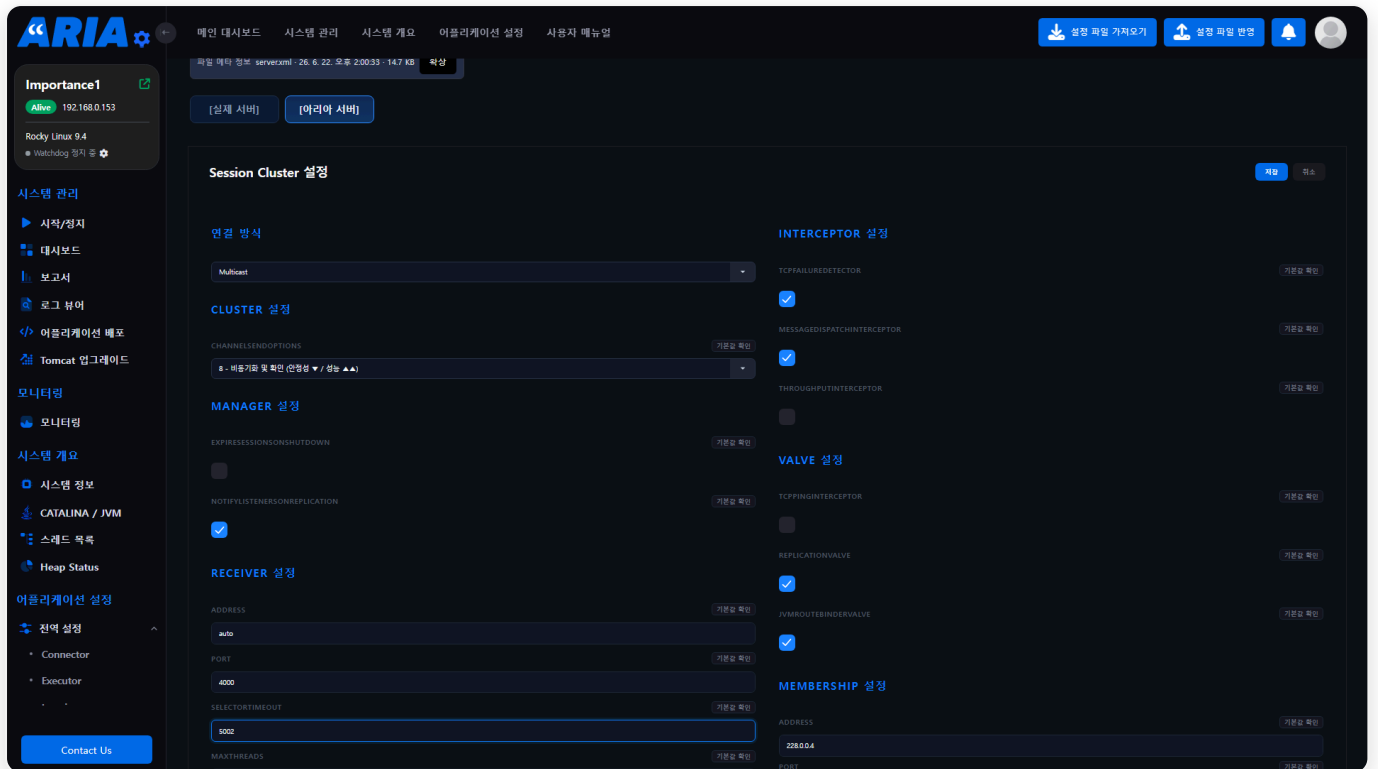
항목	설명	설정 가이드
port	멀티캐스트 통신 포트입니다.	기본 45564 . 0~65535 범위
frequency	멤버십 정보를 전송하는 주기(ms)입니다.	기본 500
dropTime	응답이 없는 노드를 클러스터에서 제거하기 전 대기 시간(ms)입니다.	기본 3000

Static 멤버

항목	설명	설정 가이드
host	정적 멤버로 추가할 노드의 호스트명 또는 IP 주소입니다.	유효한 IP 또는 도메인 입력
port	정적 멤버와 통신할 포트입니다.	0~65535 범위
uniqueId	정적 멤버를 클러스터 내에서 고유하게 식별하는 ID입니다.	노드 간 충돌 방지를 위해 고유값 지정

6.15.4 4. Cluster 설정 편집

1. [아리아 서버] 탭에서 편집할 클러스터를 선택합니다.
2. 수정 버튼을 클릭해 편집 모드로 전환합니다.



1. **설정 항목**을 참고해 연결 방식·각 구역 속성을 변경합니다.
2. **저장**을 클릭하면 아리아 작업본의 `server.xml` 에 반영되고 반영 대기 목록에 등록됩니다.

버튼	동작
수정	선택 항목 편집 모드 진입
저장	변경 내용을 아리아 작업본에 저장 (반영 대기)
취소	편집 중인 변경 사항을 되돌림
완료	신규 클러스터 등록 시 입력 내용 확정

주의: **저장**은 즉시 Tomcat에 적용되지 않습니다. 헤더의 **설정 파일 반영**으로 실제 서버에 적용해야 합니다.

6.15.5 5. 서버 반영

1. 설정을 저장하면 헤더 우측 **반영 대기** 배지에 변경 건수가 표시됩니다.
2. **설정 파일 반영** 아이콘을 클릭해 반영 대기 목록을 엽니다.
3. 대상 항목을 선택하고 **반영**을 실행합니다.

자세한 절차는 **설정 변경 반영** 문서를 참고하세요.

6.15.6 💡 활용 팁

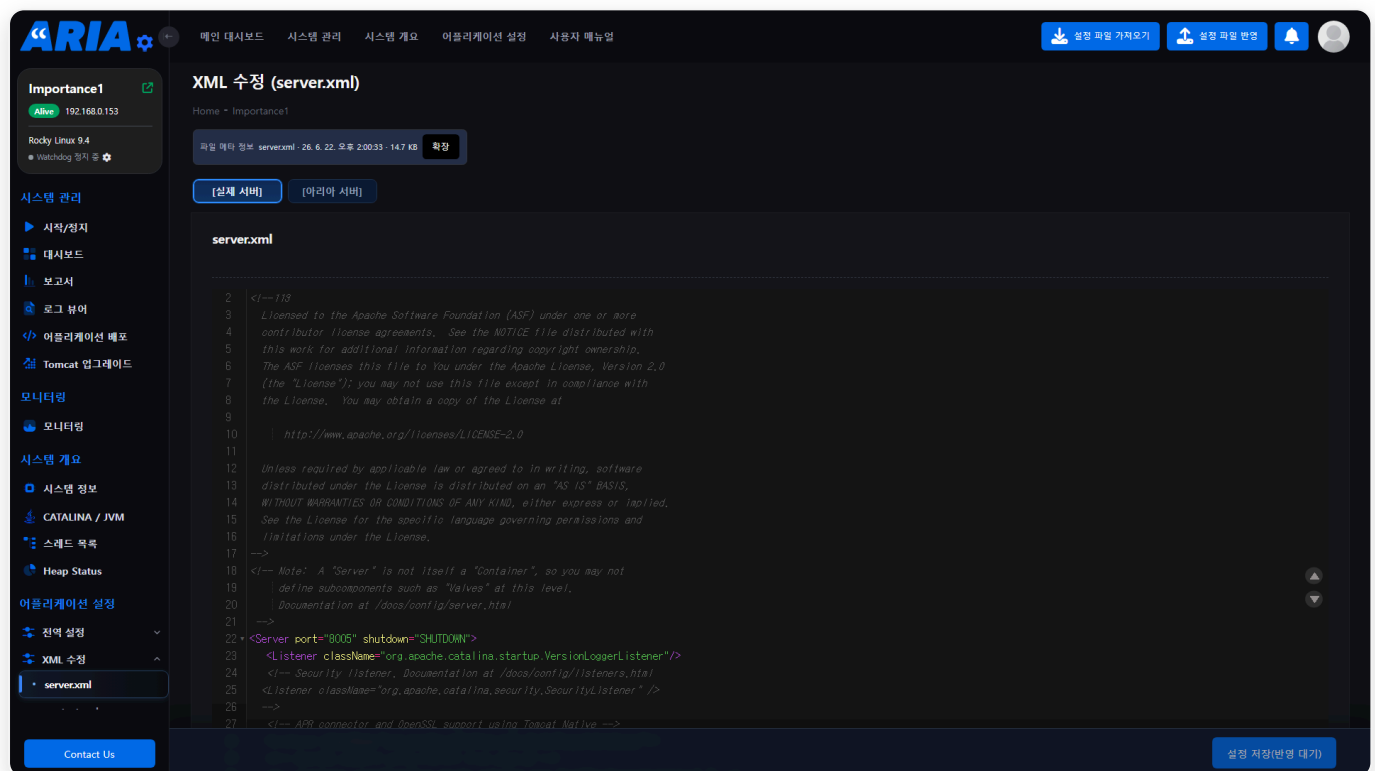
- **실제 서버와 비교:** 반영 전 [실제 서버] 탭과 [아리아 서버] 탭을 번갈아 확인해 의도한 변경인지 검증하세요.
- **설정 이력:** 페이지 제목 옆 **설정 이력** 버튼으로 `server.xml` 변경 이력을 조회·복원할 수 있습니다. ([설정 변경 이력](#))
- **가상 호스트별 클러스터:** 특정 vhost 범위 설정은 [가상 호스트 - Cluster](#)를 사용하세요.

6.16 XML 수정 - server.xml

경로: `/servers/management/config?agent_id={agent_id}&conf_type=server.xml`

Tomcat 핵심 설정 파일 **server.xml**을 에디터로 직접 편집하는 페이지입니다. Connector, Host, Cluster, GlobalNamingResources 등 서버 전역 설정을 XML로 수정할 수 있습니다.

6.16.1 화면



6.16.2 1. 화면 구성

영역	설명
설정 소스 전환	[아리아 서버] / [실제 서버] 탭
설정 파일 메타	아리아 작업본 vs 실제 <code>server.xml</code> 비교

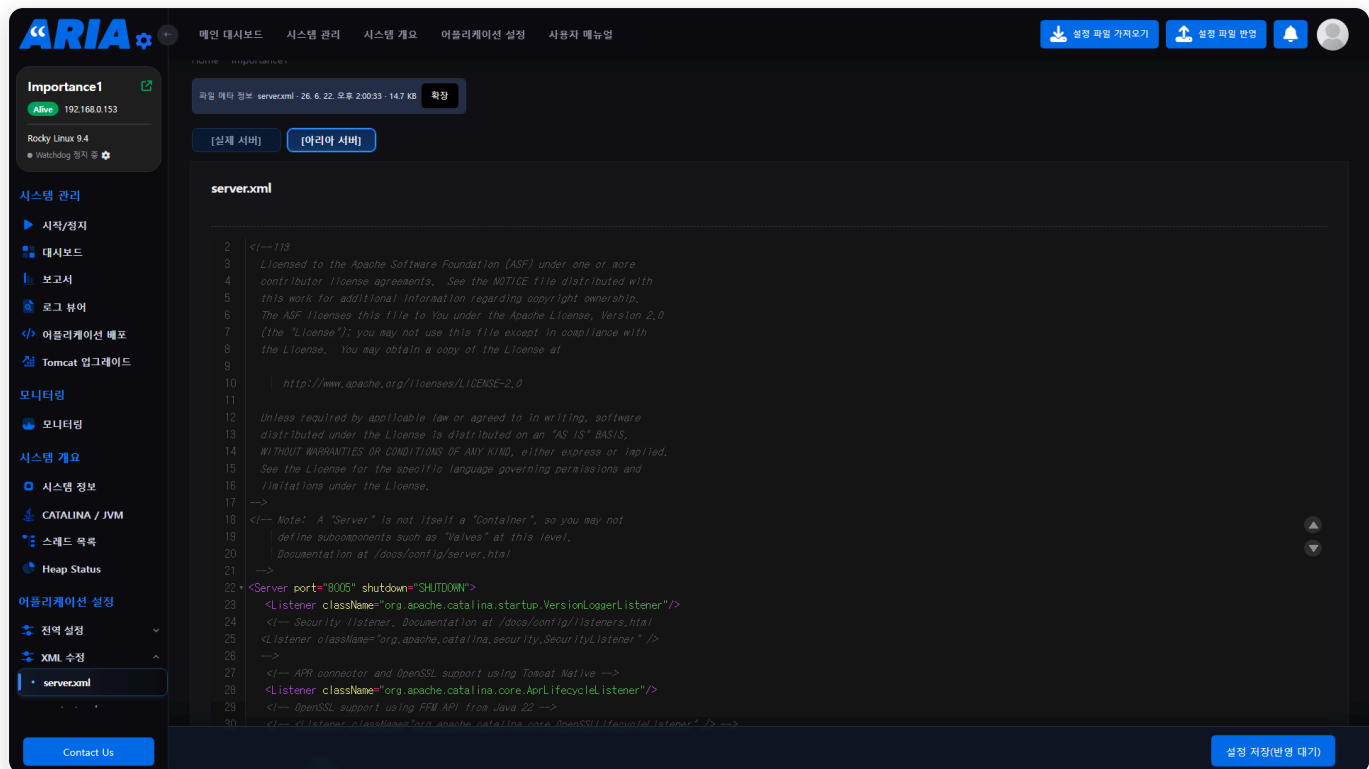
영역	설명
XML 에디터	server.xml 편집 영역
설정 저장(반영 대기)	아리아 작업본 저장 및 반영 대기 등록
설정 이력	server.xml 변경 이력 페이지로 이동

2. 설정 소스 전환

[아리아 서버] 탭은 ARIA 작업본의 server.xml 을 에디터로 직접 편집하는 화면입니다. **설정 저장(반영 대기)** 로 반영 대기 목록에 등록합니다.

[실제 서버] 탭은 Tomcat에 적용된 server.xml 을 읽기 전용으로 조회합니다.

The screenshot shows the ARIA web interface. The sidebar on the left contains various navigation links. The main content area displays the server.xml file with line numbers from 2 to 30. The file content includes XML tags for the server configuration, such as <!--|13, <!--|14, <!--|15, <!--|16, <!--|17, <!--|18, <!--|19, <!--|20, <!--|21, <!--|22, <!--|23, <!--|24, <!--|25, <!--|26, <!--|27, <!--|28, <!--|29, and <!--|30. At the bottom right of the interface, there is a button labeled '설정 저장(반영 대기)'.



원격 Tomcat 설정이 로컬 작업본과 달라진 경우, 진입 시 최신 설정을 다시 불러올지 묻는 안내가 표시될 수 있습니다.

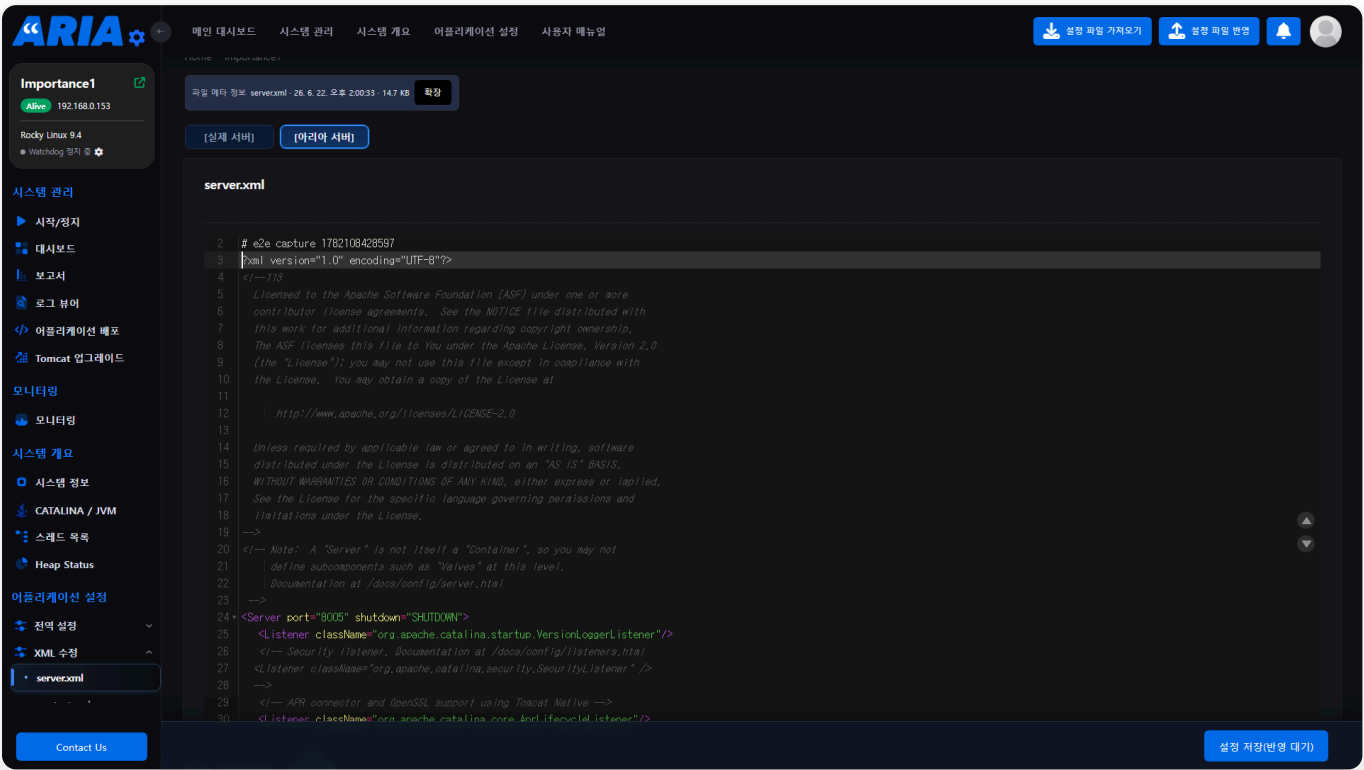
6.16.3 3. 설정 항목

server.xml 은 XML 직접 편집 방식입니다. 주요 최상위 요소는 아래와 같으며, 가능하면 전용 UI 페이지에서 편집하는 것을 권장합니다.

요소	설명	설정 가이드
<Server>	Tomcat 인스턴스 루트	port (Shutdown), shutdown 속성
<Service>	ConnectorEngine 묶음	name 속성
<Connector>	HTTP/HTTPS/AJP 리스너	Connector 설정 UI 권장
<Executor>	공유 스레드 풀	Executor 설정 UI 권장
<Engine>	요청 처리 엔진	defaultHost , jvmRoute (클러스터)
<Host>	가상 호스트	가상 호스트 - Host UI 권장
<Cluster>	세션 클러스터	Cluster 설정 UI 권장
<GlobalNamingResources>	전역 JNDI 리소스	DB Pool 설정 UI 권장

6.16.4 4. XML 편집 및 저장

1. [아리아 서버] 탭에서 에디터의 `server.xml` 내용을 수정합니다.



- 1. **설정 저장(반영 대기)** 버튼을 클릭합니다.
- 2. 저장이 완료되면 반영 대기 목록에 등록됩니다.

버튼	동작
설정 저장(반영 대기)	편집 내용을 아리아 작업본에 저장

주의: XML 직접 편집은 구문 오류 위험이 있습니다. 가능하면 전용 UI 페이지 사용을 권장합니다.

6.16.5 5. 서버 반영

설정 변경 반영 절차에 따라 Tomcat에 적용하세요.

6.16.6 💡 활용 팁

- **UI 편집 권장:** Connector-DB Pool-Cluster 등은 [Connector 설정](#), [DB Pool 설정](#), [Cluster 설정](#) 등 전용 페이지에서 편집하는 것이 안전합니다.

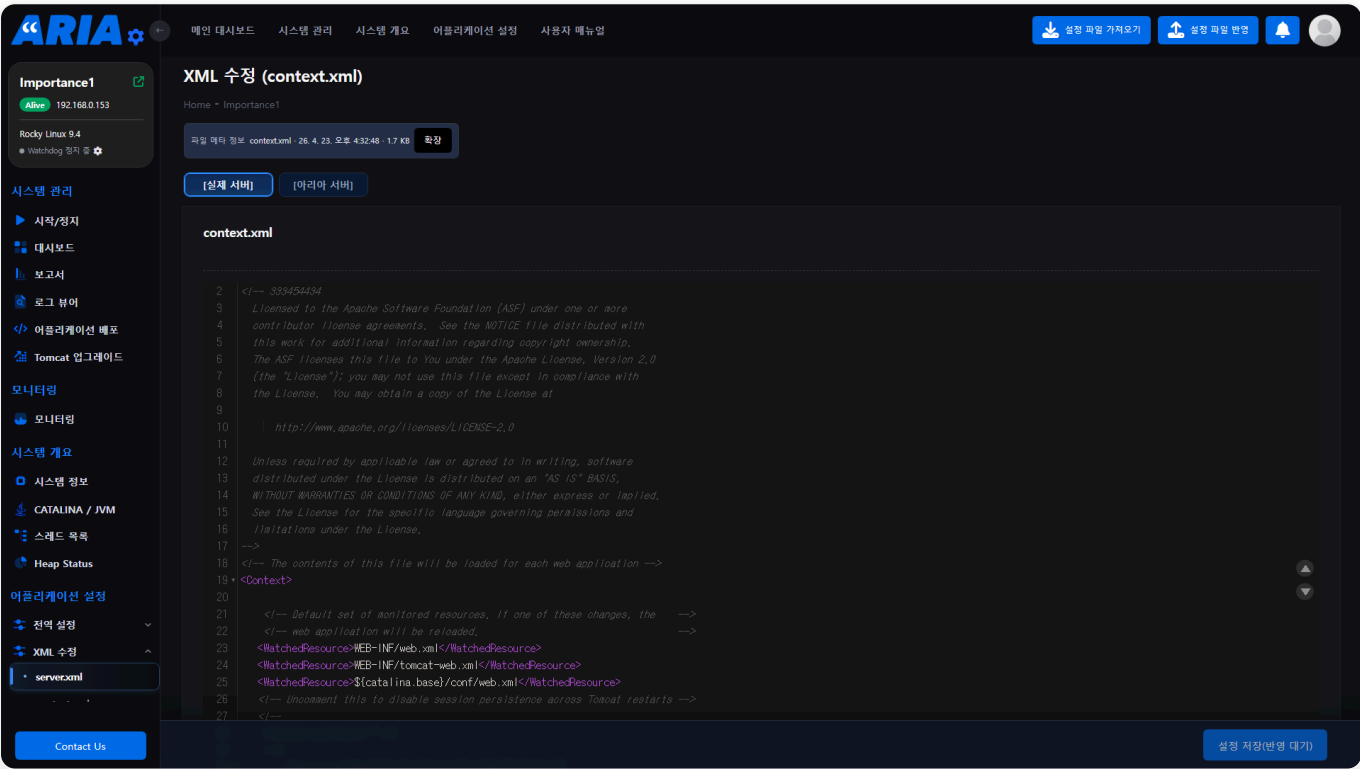
- **설정 이력**: 설정 이력 버튼으로 diff 비교·복원이 가능합니다. ([설정 변경 이력](#))

6.17 XML 수정 - context.xml

경로: `/servers/management/config?agent_id={agent_id}&conf_type=context.xml`

Tomcat 전역 **context.xml** 설정 파일을 에디터로 직접 편집하는 페이지입니다. WatchedResource, Resource, Manager 등 Context 수준 공통 설정을 XML로 수정할 수 있습니다.

6.17.1 화면



6.17.2 1. 화면 구성

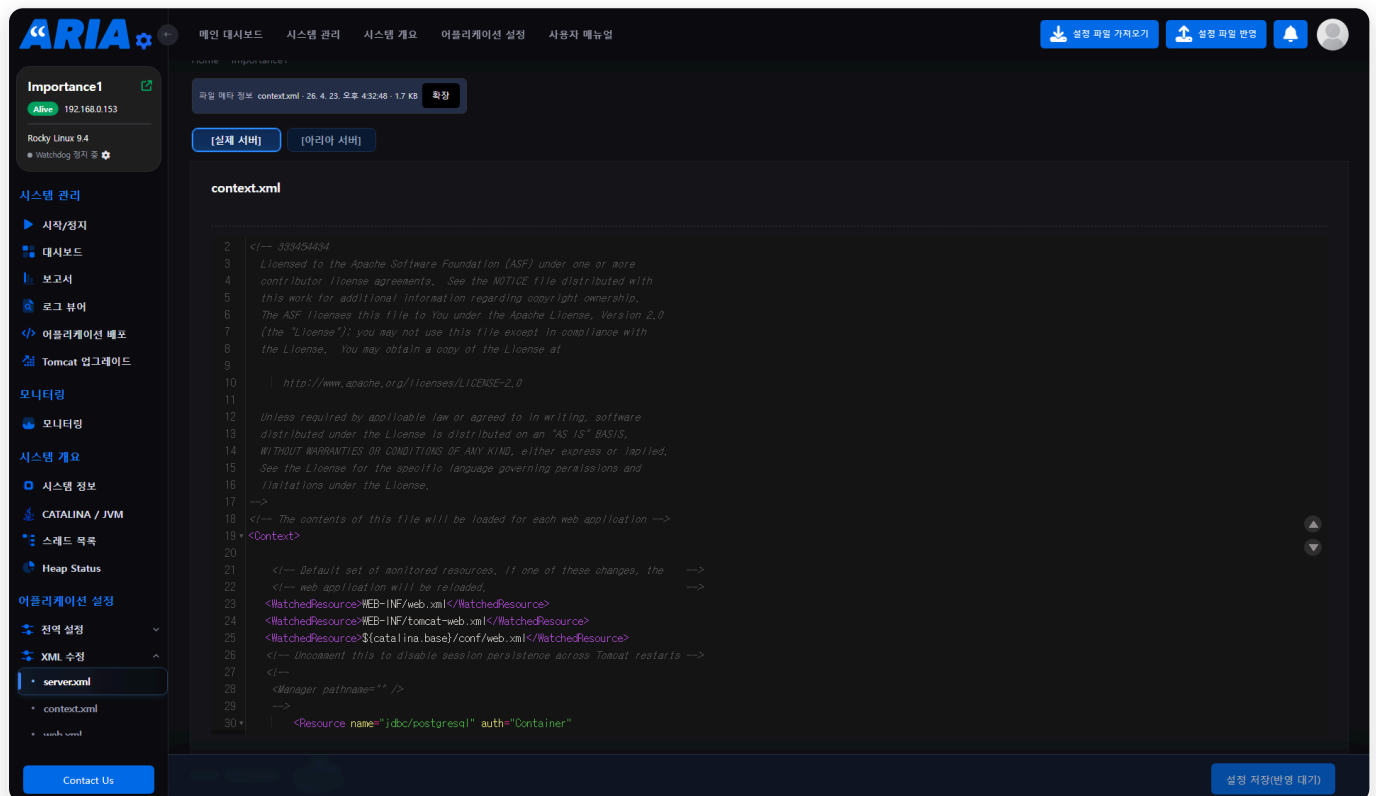
영역	설명
설정 소스 전환	[아리아 서버] / [실제 서버] 탭
설정 파일 메타	아리아 작업본과 실제 context.xml 비교
XML 에디터	context.xml 편집 영역
설정 저장(반영 대기)	아리아 작업본 저장 및 반영 대기 등록

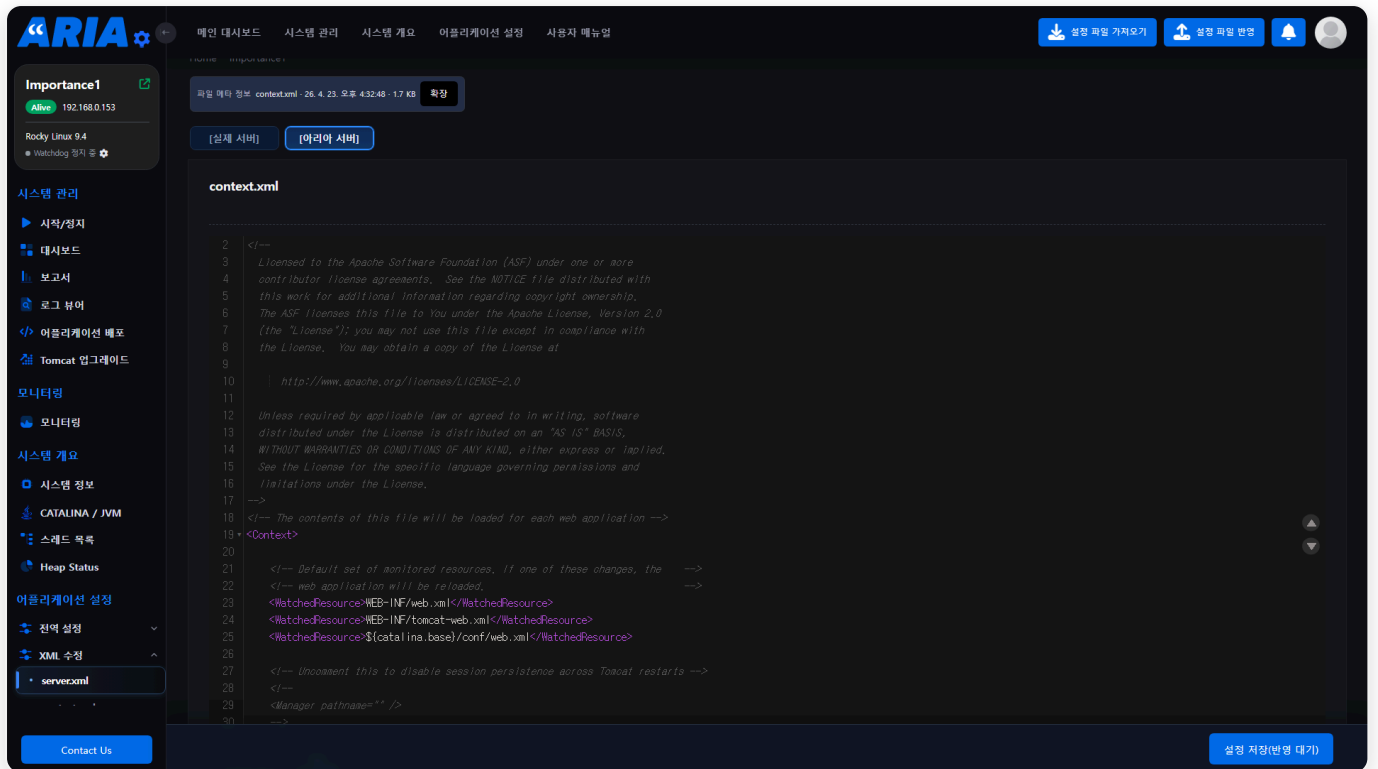
영역	설명
설정 이력	context.xml 변경 이력 페이지로 이동

2. 설정 소스 전환

[아리아 서버] 탭은 ARIA 작업본의 context.xml 을 편집하는 화면입니다. 기본으로 열립니다.

[실제 서버] 탭은 Tomcat에 적용된 context.xml 을 읽기 전용으로 조회합니다. 이 탭에서는 에디터가 읽기 전용이며 설정 저장 (반영 대기) 버튼도 비활성화됩니다.





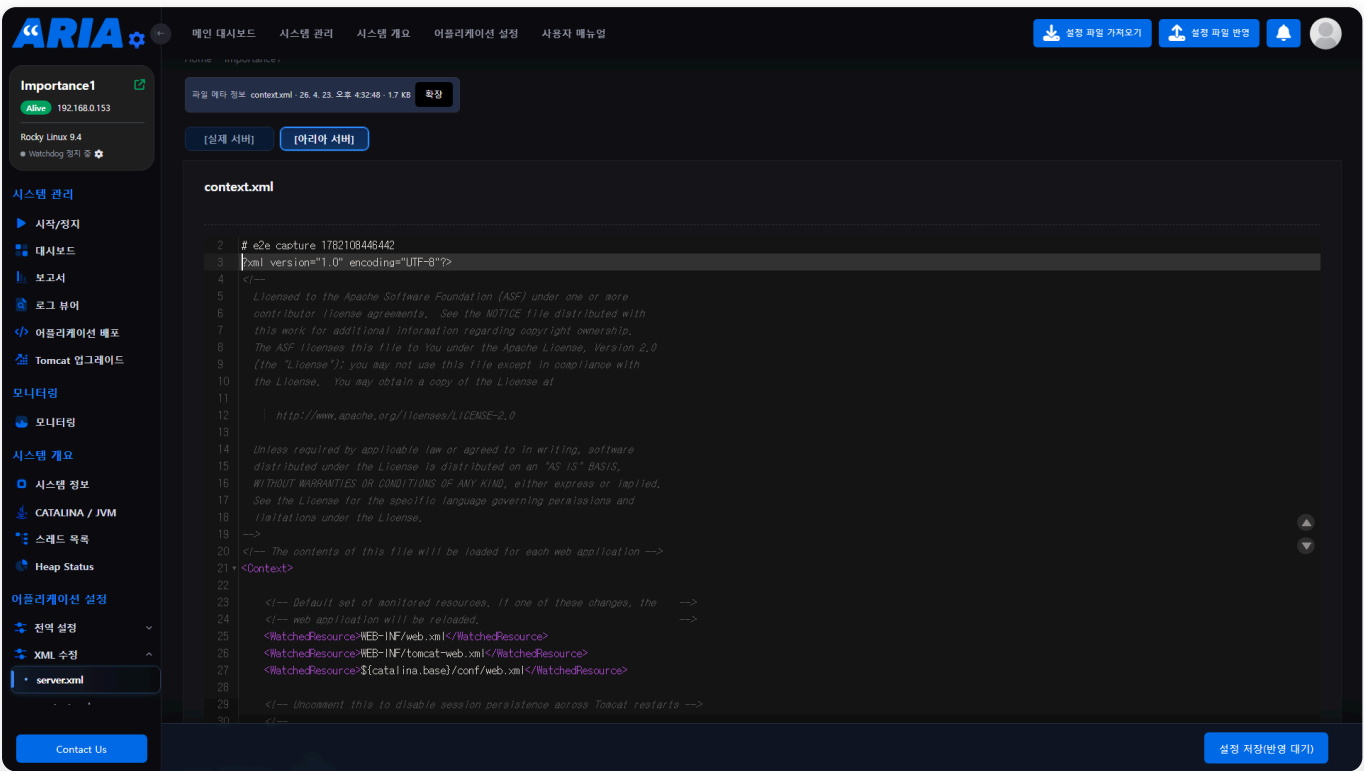
6.17.3 3. 설정 항목

전역 `context.xml` 의 `<Context>` 요소 및 하위 설정입니다. vhost별 Context는 [가상 호스트 - Context](#) UI를 사용하세요.

항목	설명	설정 가이드
docBase	웹 애플리케이션 루트 경로	WAR·디렉터리 경로. Host appBase 와 구분
reloadable	클래스 변경 시 자동 리로드	개발: <code>true</code> , 운영: <code>false</code> 권장 (성능·안정성)
crossContext	다른 Context로 dispatch 허용	<code>true</code> 시 <code>ServletContext.getContext()</code> 사용 가능
privileged	시스템 클래스 로더 접근	Manager 앱 등 특수 앱에서만 <code>true</code>
WatchedResource	변경 감시 리소스	<code>WEB-INF/web.xml</code> 등 — 변경 시 Context 리로드
Resource	JNDI 리소스 정의	DataSource 등. 전역 풀은 DB Pool 설정 참고
Manager	세션 관리자	분산 세션·영속화 설정
Loader	클래스 로더	<code>delegate</code> 등 고급 설정

6.17.4 4. XML 편집 및 저장

1. [아리아 서버] 탭에서 에디터의 context.xml 내용을 수정합니다.



- 1. 하단 **설정 저장(반영 대기)** 버튼을 클릭합니다.
- 2. 저장이 완료되면 아리아 작업본에 반영되고 반영 대기 목록에 등록됩니다.

버튼	동작
설정 저장(반영 대기)	편집 내용을 아리아 작업본에 저장
(실제 서버 탭)	저장 불가, 조회 전용

주의: 저장만으로 Tomcat에 즉시 적용되지 않습니다. 원격 설정이 변경된 경우 진입 시 drift 안내가 표시될 수 있습니다.

6.17.5 5. 서버 반영

- 1. 헤더 **반영 대기** 배지를 확인합니다.
- 2. **설정 파일 반영**에서 context.xml 항목을 선택해 반영합니다.

설정 변경 반영 문서를 참고하세요.

6.17.6 💡 활용 팁

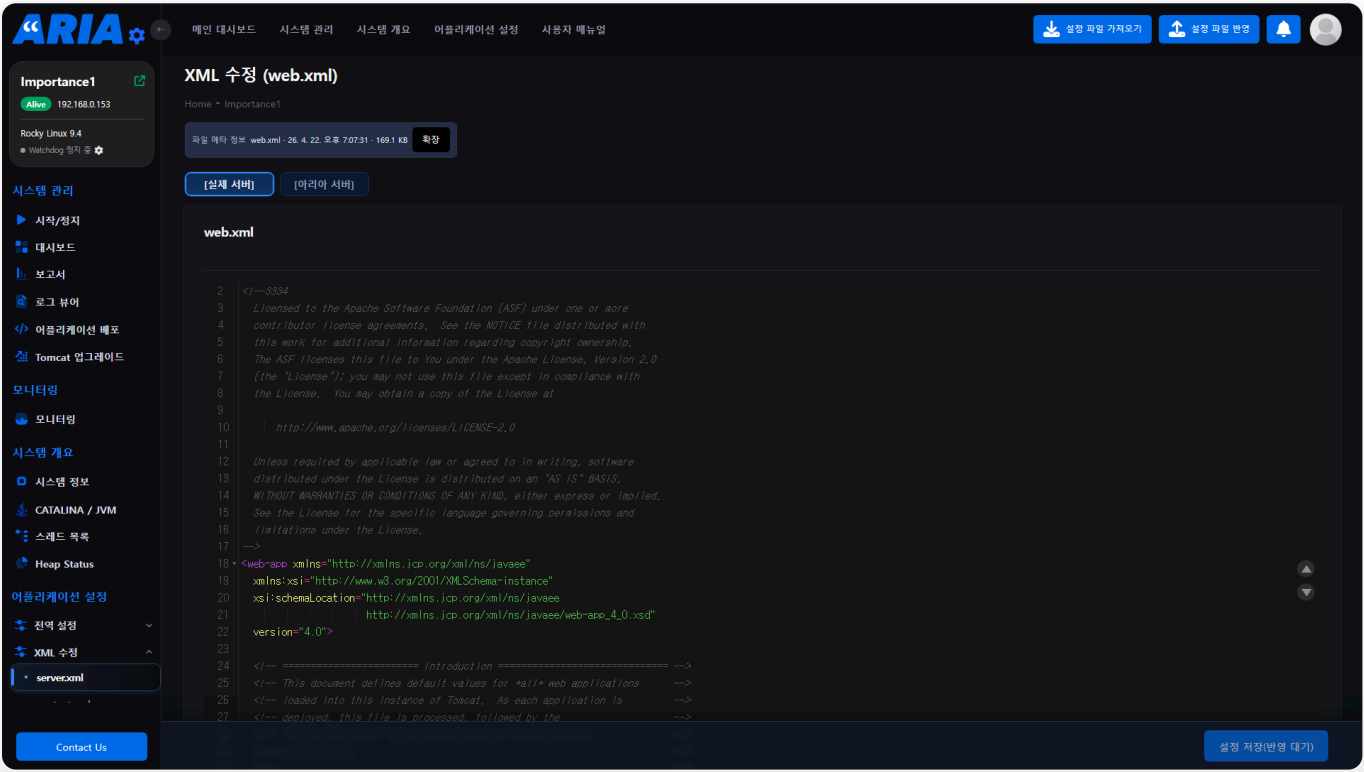
- **폼 기반 편집:** vhost별 Context는 가상 호스트 - Context에서 UI로 편집할 수 있습니다.
- **설정 이력:** 설정 이력 버튼으로 이전 버전 diff·복원이 가능합니다. (설정 변경 이력)

6.18 XML 수정 - web.xml

경로: /servers/management/config?agent_id={agent_id}&conf_type=web.xml

Tomcat **web.xml** 설정 파일을 에디터로 직접 편집하는 페이지입니다. 기본 서블릿, 필터, 세션 타임아웃 등 웹 애플리케이션 공통 배포 서술자를 수정할 수 있습니다.

6.18.1 화면



6.18.2 1. 화면 구성

영역	설명
설정 소스 전환	[아리아 서버] / [실제 서버] 탭
설정 파일 메타	아리아 작업본 vs 실제 web.xml 비교

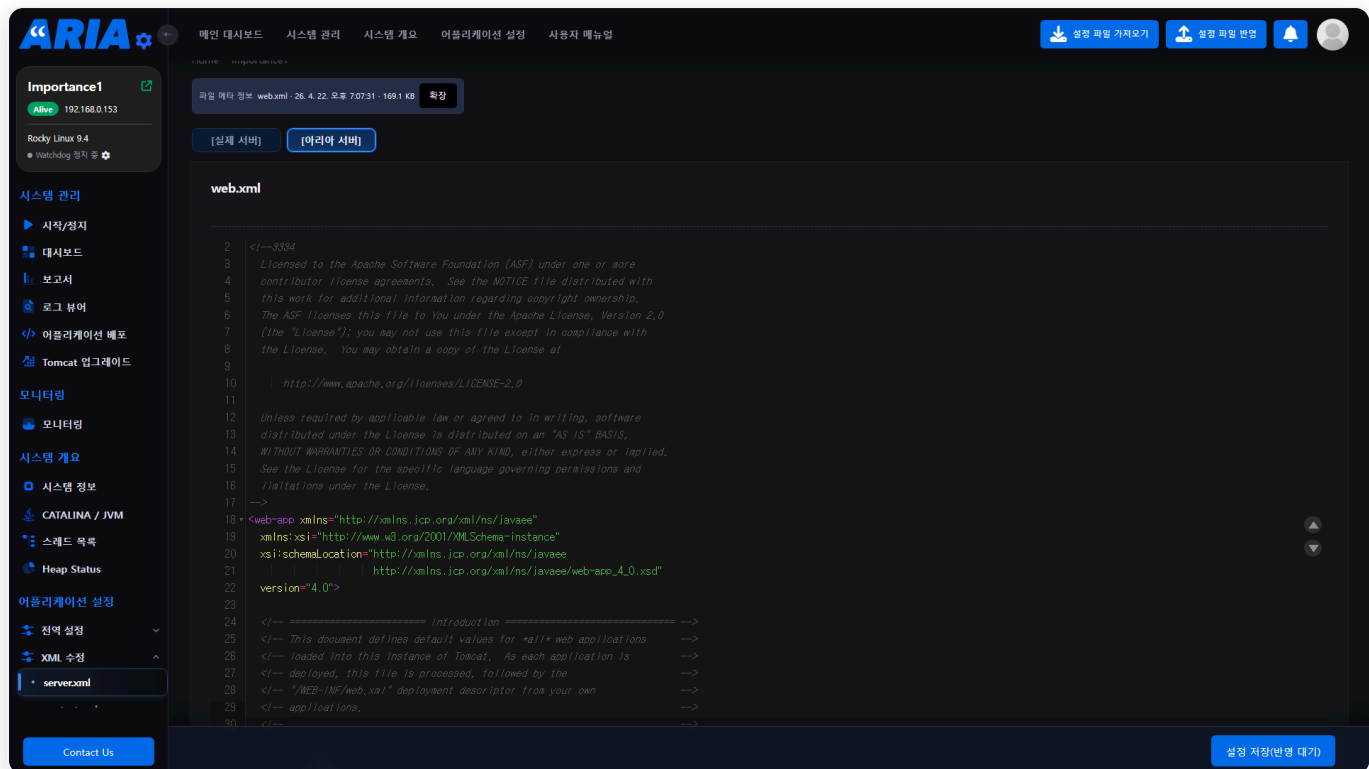
영역	설명
XML 에디터	web.xml 편집 영역
설정 저장(반영 대기)	아리아 작업본 저장 및 반영 대기 등록
설정 이력	web.xml 변경 이력 페이지로 이동

2. 설정 소스 전환

[아리아 서버] 탭은 ARIA 작업본의 web.xml 을 에디터로 편집하는 화면입니다.

[실제 서버] 탭은 Tomcat에 적용된 web.xml 을 읽기 전용으로 조회합니다.

The screenshot displays the ARIA web management interface. The top navigation bar includes links for '메인 대시보드', '시스템 관리', '시스템 개요', '어플리케이션 설정', and '사용자 매뉴얼'. On the right, there are buttons for '설정 파일 가져오기', '설정 파일 변경', and a notification bell. The left sidebar contains a '시스템 관리' section with links for '시작/종지', '대시보드', '로그 뷰어', '어플리케이션 배포', 'Tomcat 업그레이드', '모니터링', '시스템 개요', '시스템 정보', 'CATALINA / JVM', '스레드 목록', 'Heap Status', and '어플리케이션 설정'. Under '어플리케이션 설정', there is a tree view with 'server.xml', 'context.xml', and 'web.xml'. The main area shows the 'web.xml' file being edited. The file content includes XML declarations for namespaces and schema locations, followed by comments and XML elements. The interface also shows a status bar at the bottom with 'Contact Us' and '설정 저장(반영 대기)' buttons.



6.18.3 3. 설정 항목

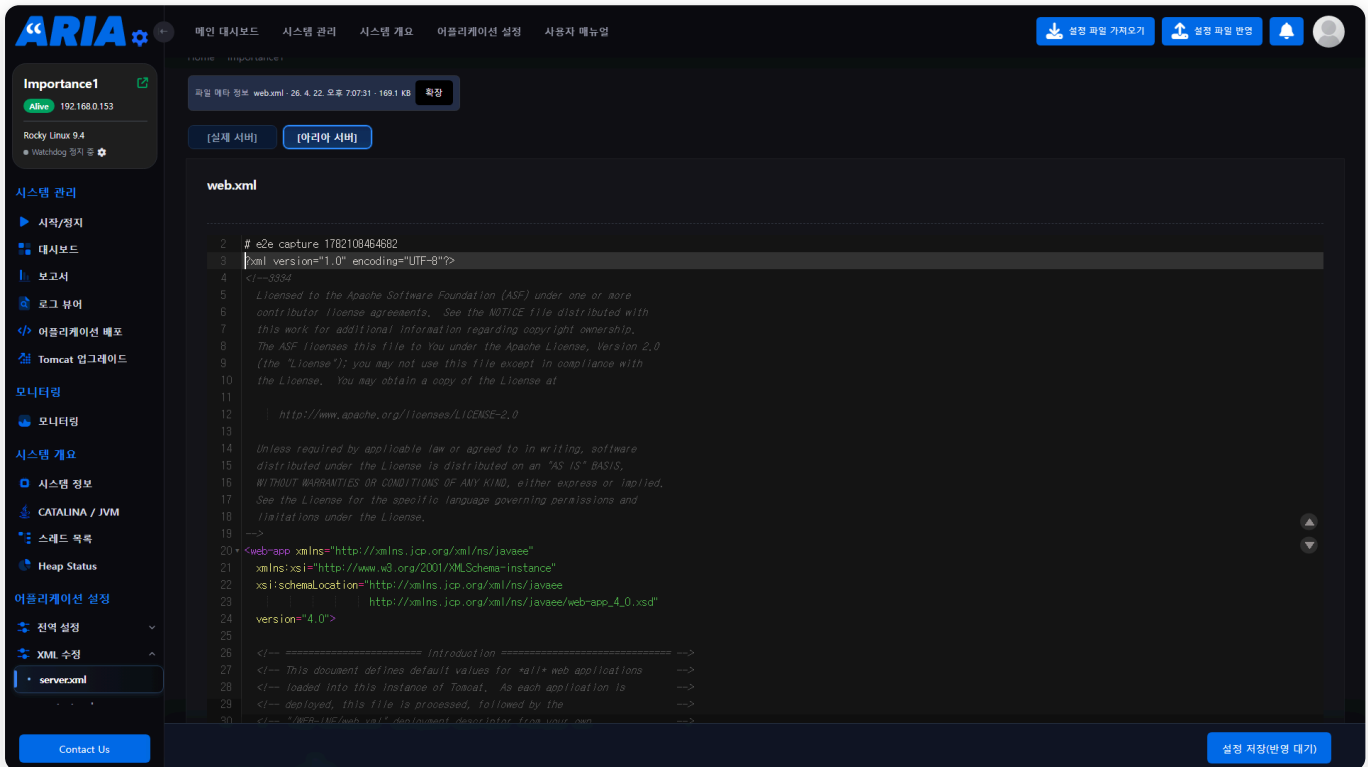
Tomcat 기본 `conf/web.xml` 의 주요 배포 서술자입니다.

요소	설명	설정 가이드
<code><servlet></code>	기본·내장 서블릿 정의	<code>servlet-name</code> , <code>servlet-class</code> , <code>init-param</code>
<code><servlet-mapping></code>	URL 패턴과 서블릿 매핑	<code>url-pattern</code> — 기본 서블릿·JSP 서블릿 등
<code><filter></code>	공통 필터 정의	<code>filter-name</code> , <code>filter-class</code>
<code><filter-mapping></code>	필터 적용 URL·서블릿	<code>url-pattern</code> 또는 <code>servlet-name</code>
<code><session-config></code>	세션 공통 설정	<code><session-timeout></code> (분) — 기본 세션 만료 시간
<code><welcome-file-list></code>	기본 환영 파일	<code>index.html</code> , <code>index.jsp</code> 등
<code><mime-mapping></code>	MIME 타입 매핑	확장자별 Content-Type
<code><error-page></code>	HTTP 오류 페이지	<code>error-code</code> 또는 <code>exception-type</code>
<code><security-constraint></code>	URL 보안 제약	인증·역할 기반 접근 제어

요소	설명	설정 가이드
<login-config>	로그인 방식	BASIC, FORM 등

6.18.4 4. XML 편집 및 저장

1. [아리아 서버] 탭에서 에디터의 web.xml 내용을 수정합니다.



1. 설정 저장(반영 대기) 버튼을 클릭합니다.
2. 저장이 완료되면 반영 대기 목록에 등록됩니다.

버튼	동작
설정 저장(반영 대기)	편집 내용을 아리아 작업본에 저장

6.18.5 5. 서버 반영

설정 변경 반영 절차에 따라 적용하세요.

6.18.6 💡 활용 팁

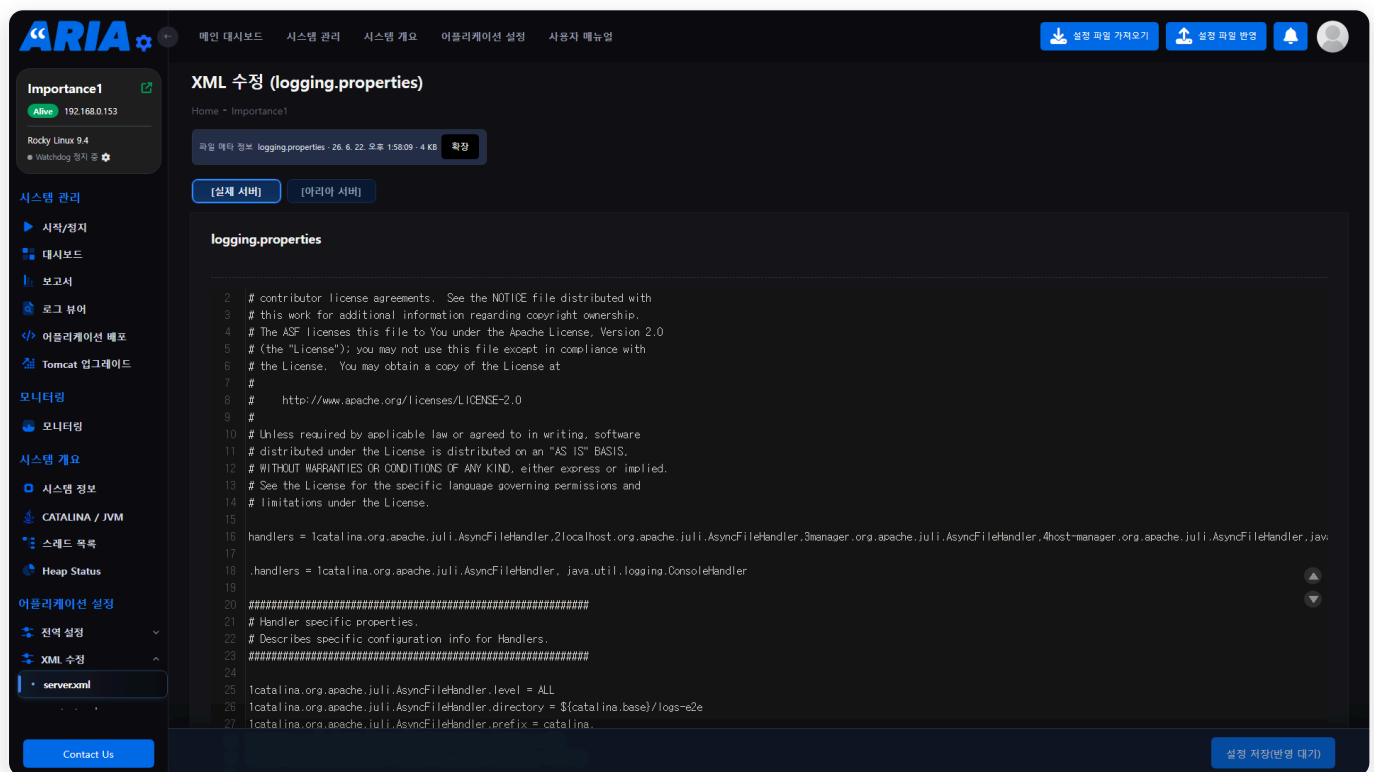
- **배포 연계:** WAR 배포·리로드는 **어플리케이션 배포**에서 제어할 수 있습니다.
- **설정 이력:** **설정 이력** 버튼으로 이전 버전을 비교·복원할 수 있습니다.

6.19 XML 수정 - logging.properties

경로: `/servers/management/config?agent_id={agent_id}&conf_type=logging.properties`

Tomcat **logging.properties** 파일을 에디터로 직접 편집하는 페이지입니다. 핸들러·Appender·로그 레벨 등을 properties 형식으로 자유롭게 수정할 수 있습니다.

6.19.1 화면



6.19.2 1. 화면 구성

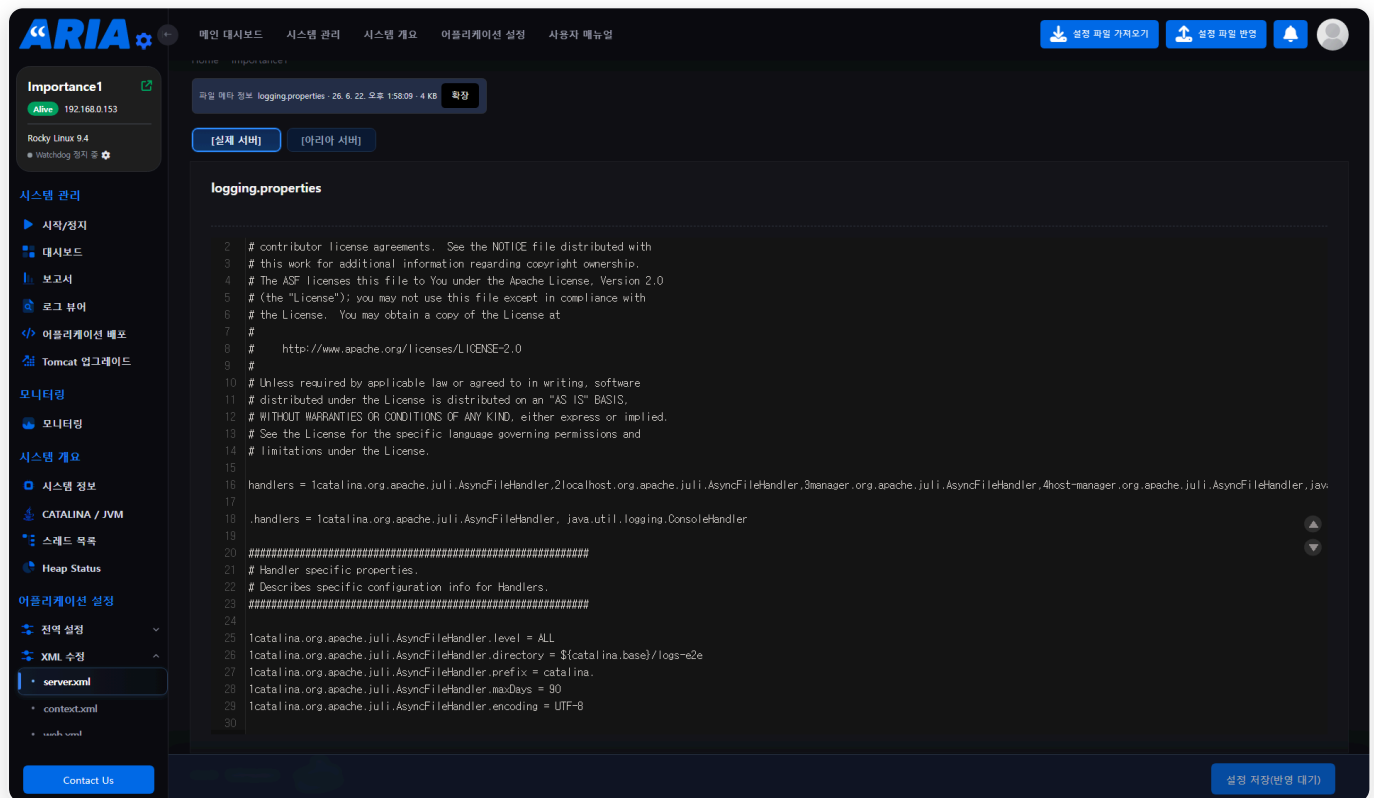
영역	설명
설정 소스 전환	[아리아 서버] / [실제 서버] 탭
설정 파일 메타	아리아 작업본과 실제 <code>logging.properties</code> 비교

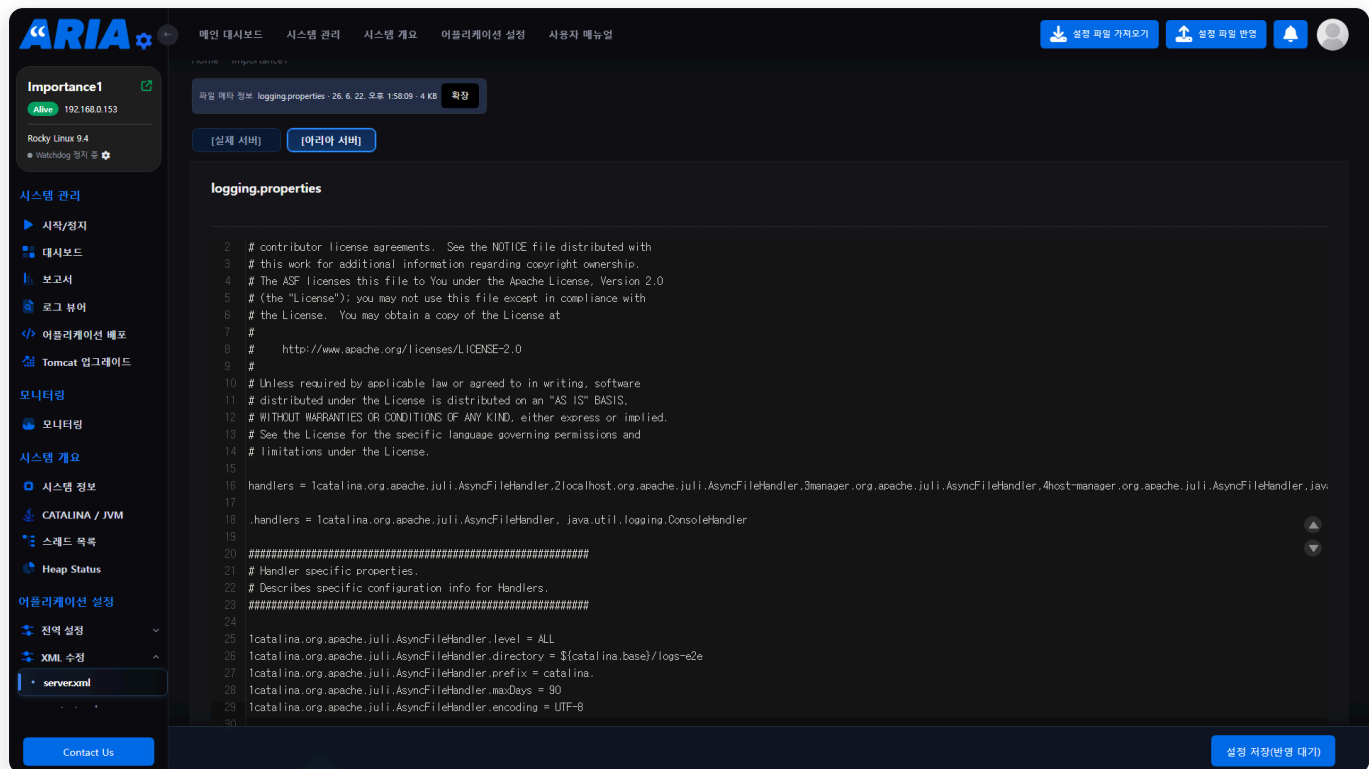
영역	설명
Properties 에디터	편집 영역
설정 저장(반영 대기)	아리아 작업본 저장 및 반영 대기 등록
설정 이력	<code>logging.properties</code> 변경 이력 페이지로 이동

2. 설정 소스 전환

[아리아 서버] 탭은 ARIA 작업본의 `logging.properties` 를 에디터로 편집하는 화면입니다.

[실제 서버] 탭은 Tomcat에 적용된 파일을 읽기 전용으로 조회합니다. 이 탭에서는 에디터가 읽기 전용이며 **설정 저장(반영 대기)** 버튼이 비활성화됩니다.





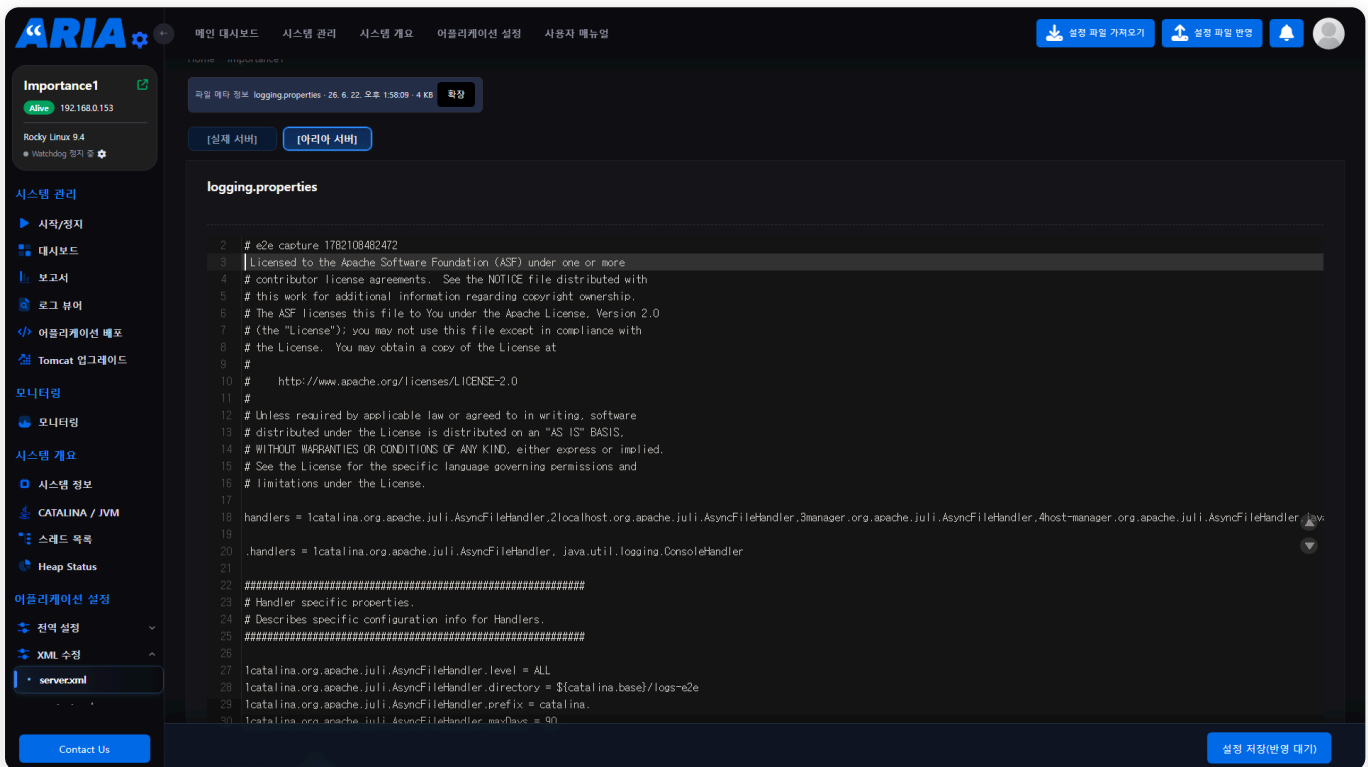
6.19.3 3. 설정 항목

항목 (키 패턴)	설명	설정 가이드
<code>handlers</code>	사용할 핸들러 목록	심표로 구분 (예: <code>1catalina.org.apache.juli.AsyncFileHandler</code>)
<code>.{category}.handlers</code>	카테고리별 핸들러	<code>catalina</code> , <code>localhost</code> , <code>manager</code> 등
<code>.{category}.level</code>	카테고리별 로그 레벨	<code>INFO</code> , <code>WARNING</code> , <code>FINE</code> 등. Logging 설정 UI로 편집 권장
<code>.{category}.useParentHandler</code>	상위 핸들러 상속	<code>false</code> 시 해당 카테고리만 별도 출력
<code>*.directory</code>	로그 파일 디렉터리	<code>\${catalina.base}/logs</code> 등
<code>*.prefix</code>	로그 파일 접두사	카테고리별 구분
<code>*.suffix</code>	로그 파일 접미사	<code>.log</code> 등
<code>*.encoding</code>	파일 인코딩	<code>UTF-8</code> 권장

항목 (키 패턴)	설명	설정 가이드
*.maxDays	로그 보관 일수	숫자 (일)
*.bufferSize	버퍼 크기	Async 핸들러 성능 튜닝

6.19.4 4. 파일 편집 및 저장

1. [아리아 서버] 탭에서 에디터 내용을 수정합니다.



1. 설정 저장(반영 대기) 버튼을 클릭합니다.
2. 저장이 완료되면 반영 대기 목록에 등록됩니다.

버튼	동작
설정 저장(반영 대기)	편집 내용을 아리아 작업본에 저장

6.19.5 5. 서버 반영

설정 변경 반영 절차에 따라 적용하세요.

6.19.6 💡 활용 팁

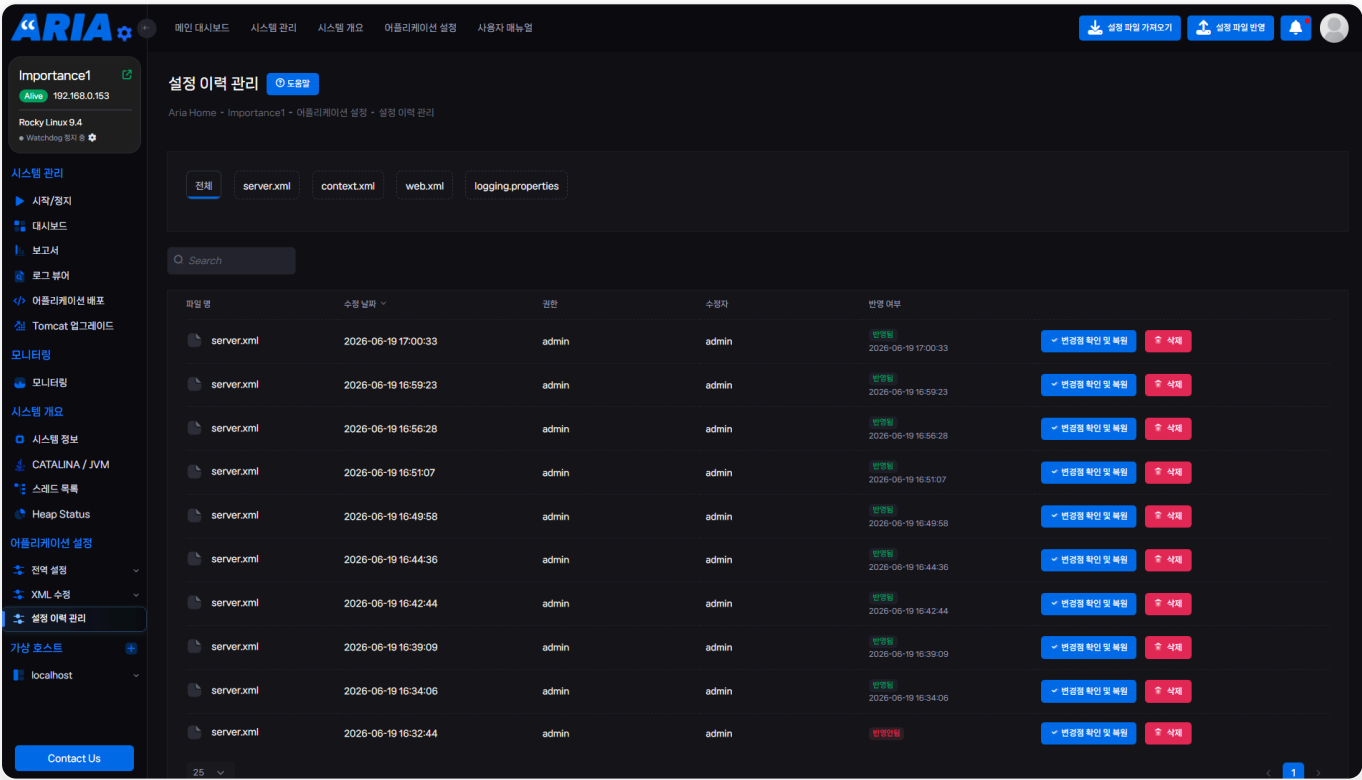
- **폼 기반 편집**: 카테고리별 Level만 변경할 때는 **Logging 설정**이 더 편리할 수 있습니다.

6.20 설정 변경 이력

경로: `/servers/management/history?agent_id={agent_id}`

ARIA에서 편집-저장된 Tomcat 설정 파일(`server.xml` , `context.xml` , `web.xml` , `logging.properties`)의 **변경 이력**을 조회하는 페이지입니다. 이력별 변경 전·후 diff를 비교하고, 이전 버전으로 아리아 작업본을 복원할 수 있습니다.

6.20.1 화면



6.20.2 1. 화면 구성

영역	설명
파일 유형 탭	전체 / server.xml / context.xml / web.xml / logging.properties 필터
검색	파일명·수정자 등 키워드 검색
이력 목록	저장된 설정 변경 이력 테이블

영역	설명
변경점 확인 및 복원	선택 이력의 diff 비교 및 복원
삭제	불필요한 이력 레코드 제거

이력 목록 컬럼

컬럼	설명
파일 명	변경된 설정 파일 종류
수정 날짜	이력이 저장된 시각
권한	수정 당시 사용자 권한
수정자	변경을 수행한 계정
반영 여부	반영됨 (녹색) / 반영안됨 (빨간색) 및 반영 시각

2. 파일 유형 필터

상단 탭으로 조회 대상 파일을 필터링합니다.

탭	설명
전체	모든 설정 파일 이력 표시
server.xml	Connector, Host, Cluster 등 서버 전역 설정 이력
context.xml	Context 수준 설정 이력
web.xml	웹 애플리케이션 배포 서술자 이력
logging.properties	로깅 핸들러·레벨 설정 이력

검색창에 파일명을 입력해도 동일하게 필터링됩니다. 전역 설정·가상 호스트 설정 페이지의 **설정 이력** 버튼으로 진입하면 해당 파일 유형이 자동으로 필터됩니다.

6.20.3 3. 변경점 확인 및 복원

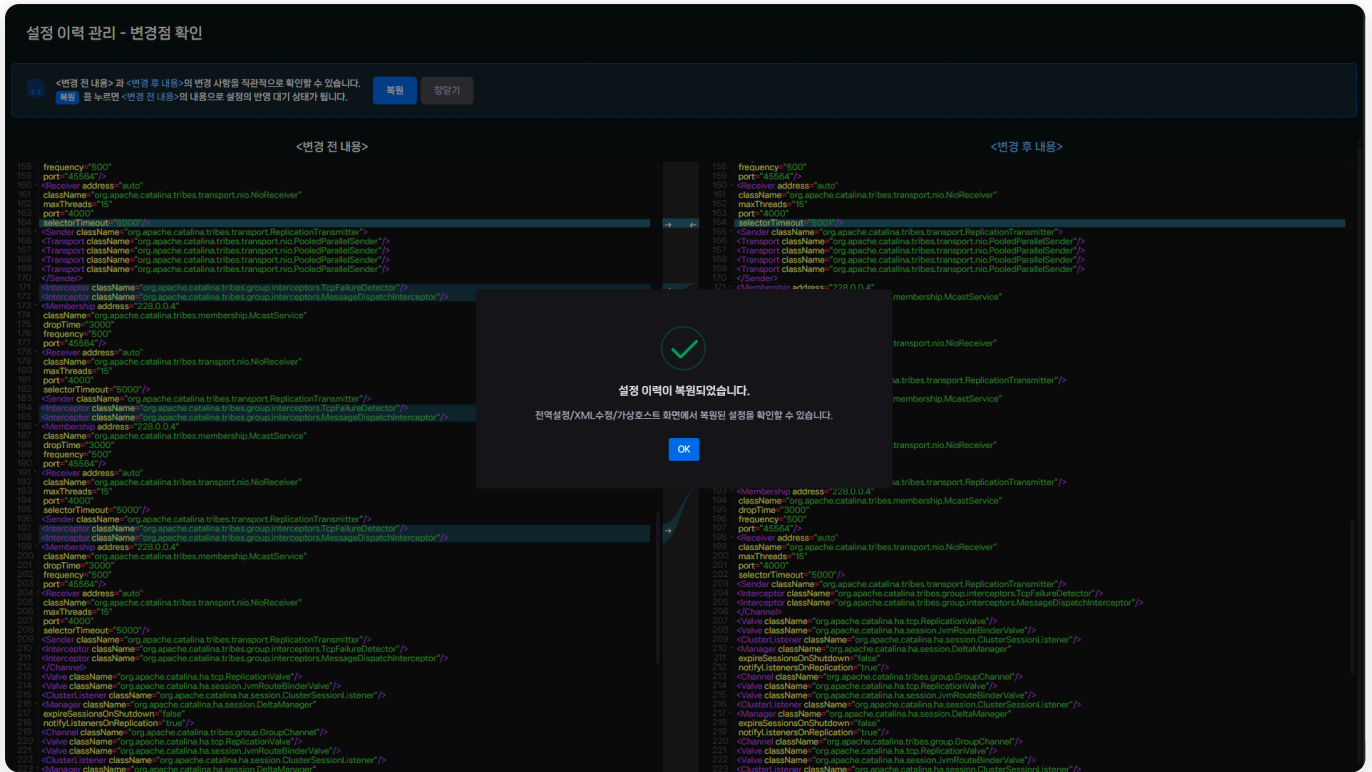
1. 복원할 이력 행의 **변경점 확인** 및 **복원** 버튼을 클릭합니다.
2. 새 창에서 **변경 전 내용**과 **변경 후 내용**을 나란히 비교합니다.

[illegible]

1. 복원 버튼을 클릭하고 확인 모달에서 확정합니다.

[illegible]

1. 복원이 완료되면 아리아 작업본이 해당 이력의 **변경 전 내용**으로 되돌아갑니다.



버튼	동작
복원	선택 이력의 변경 전 내용으로 아리아 작업본 복원 (반영 대기 상태)
창닫기	diff 창 닫기

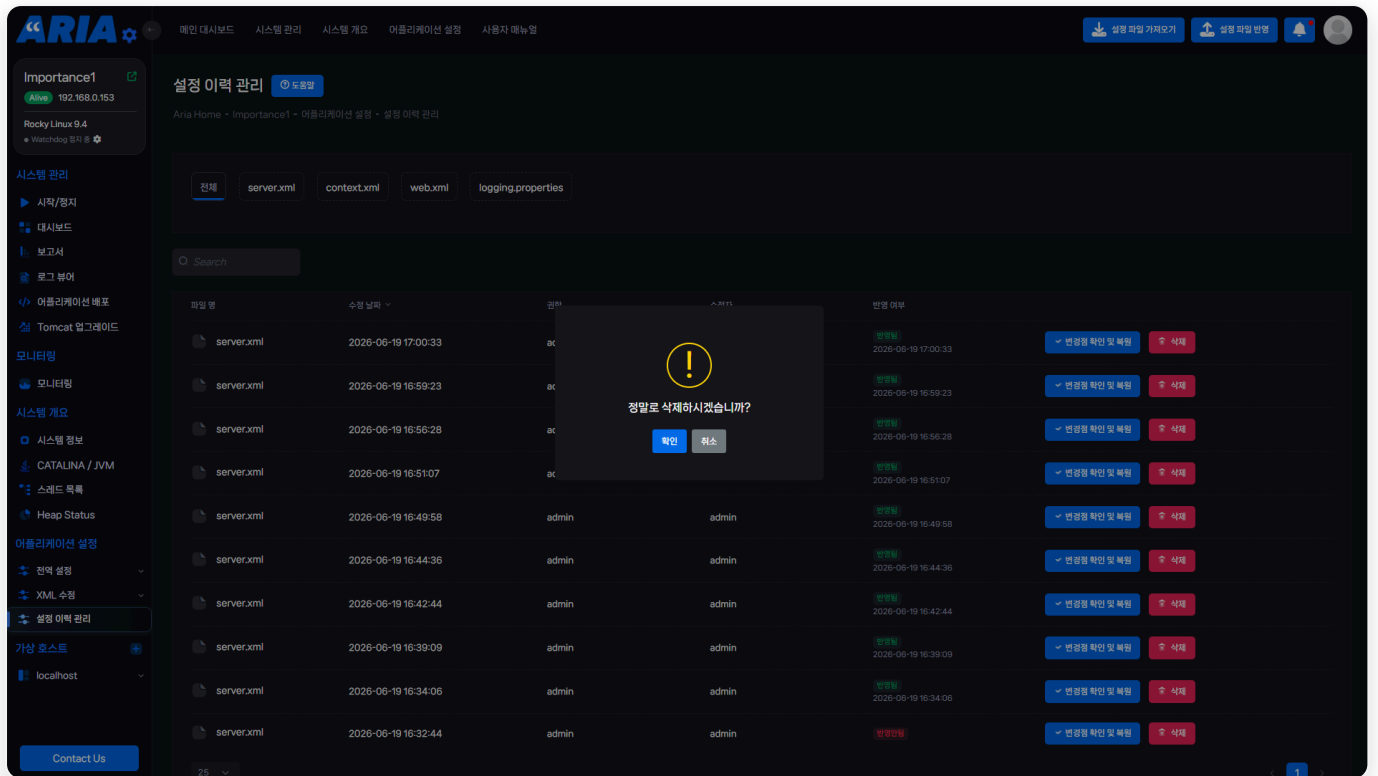
주의: 복원은 설정 파일 가져오기와 같이 로컬 작업본만 변경합니다. Tomcat에 즉시 적용되지 않으므로, 헤더 설정 파일 반영으로 서버에 반영해야 합니다.

원격 서버 설정이 로컬 작업본보다 최신인 경우, 복원 전에 최신 원격 설정을 다시 로드할지 묻는 안내가 표시될 수 있습니다.

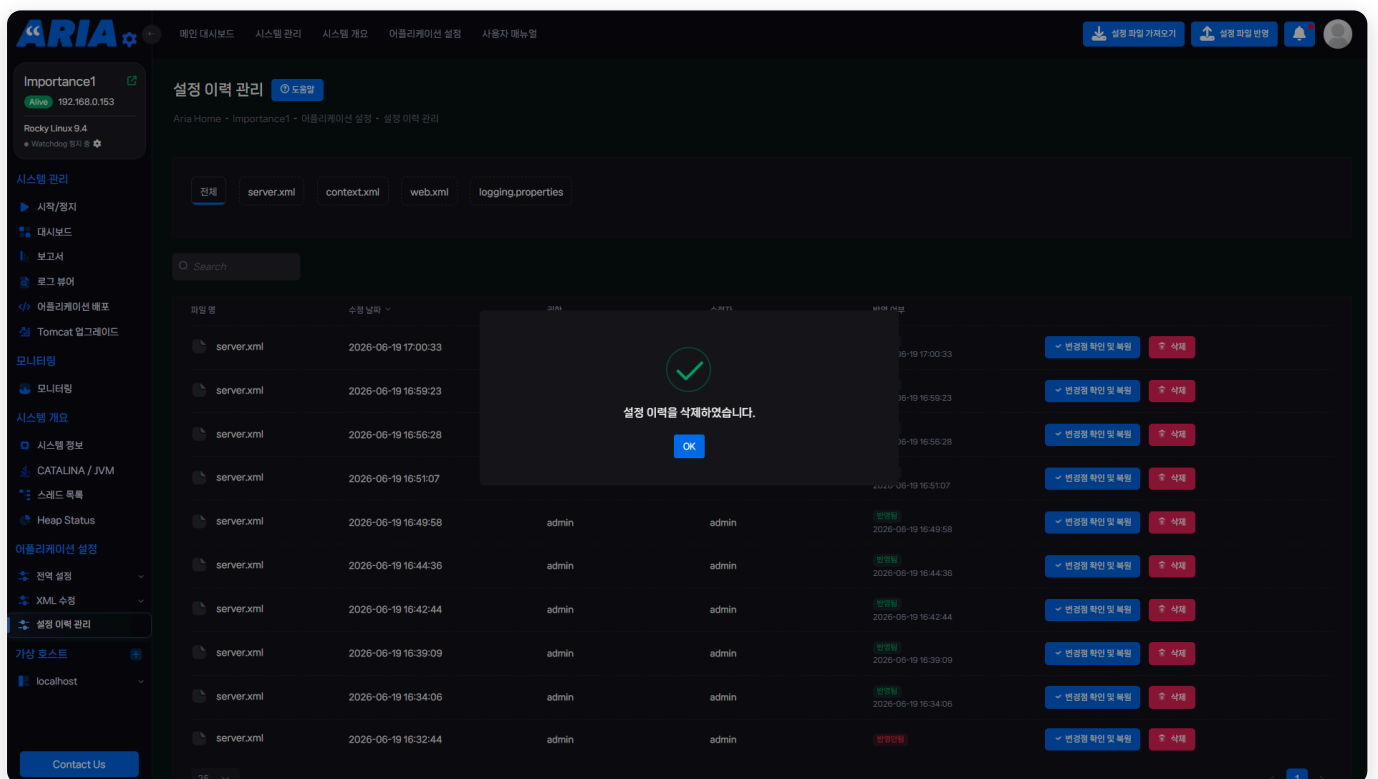
6.20.4 4. 이력 삭제

더 이상 필요 없는 이력 레코드를 삭제합니다.

1. 삭제할 행의 **삭제** 버튼을 클릭합니다.
2. 확인 modal에서 **확인**을 클릭합니다.



1. 삭제가 완료되면 목록이 갱신됩니다.



주의: 이력 삭제는 DB에 저장된 기록만 제거하며, 현재 Tomcat에 적용된 설정에는 영향을 주지 않습니다.

6.20.5 5. 서버 반영

복원 후에는 전역 설정·XML 수정·가상 호스트 화면에서 복원된 내용을 확인한 뒤, 헤더 **설정 파일 반영**으로 **설정 변경 반영** 절차를 진행하세요.

6.20.6 💡 활용 팁

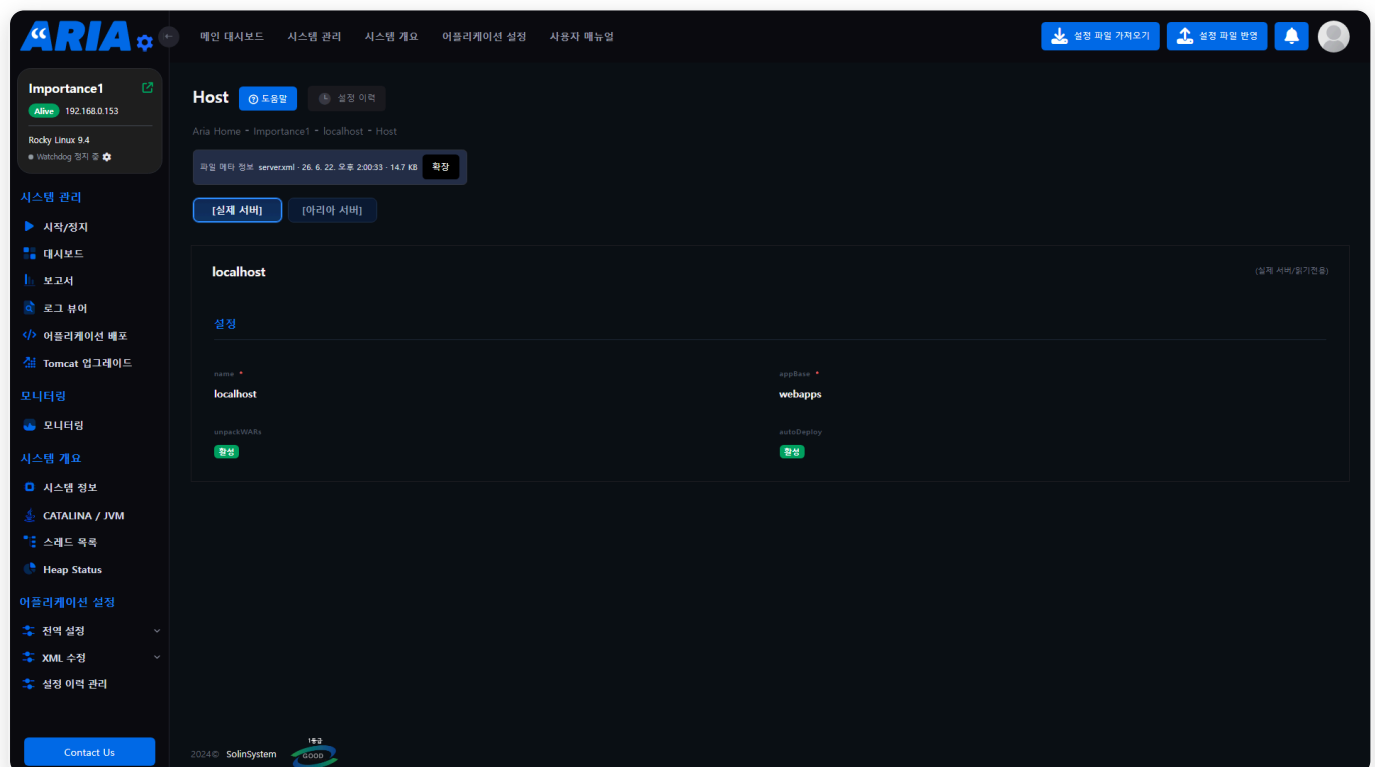
- **반영 여부 확인:** 반영안됨 이력은 저장만 되고 서버에 적용되지 않은 상태입니다. 반영 시각이 있는 반영됨 이력이 실제 서버에 적용된 버전입니다.
- **설정 페이지 연계:** Connector 설정, server.xml 등 각 설정 페이지의 **설정 이력** 버튼으로 해당 파일 이력만 바로 조회할 수 있습니다.
- **롤백 절차:** 문제 발생 시 → 이력에서 변경점 확인 → 복원 → 설정 화면에서 내용 검증 → 설정 파일 반영.

6.21 가상 호스트 - Host

경로: /servers/management/config/vhost/host?agent_id={agent_id}&vhost={vhost}

선택한 가상 호스트의 **Host** 기본 속성(name , appBase , unpackWARs , autoDeploy 등)을 웹 UI로 편집하는 페이지입니다. 변경 내용은 아리아 작업본의 server.xml 에 저장되며, 헤더 **설정 파일 반영**으로 Tomcat에 적용합니다.

6.21.1 화면



6.21.2 1. 화면 구성

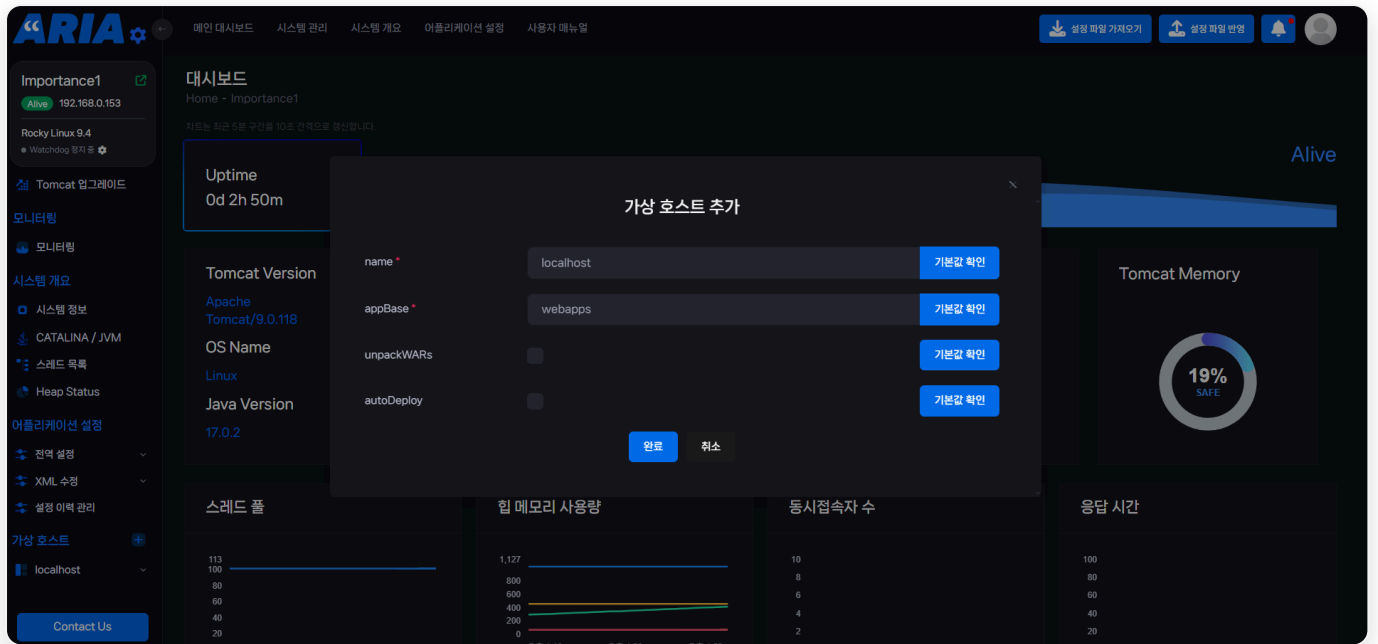
영역	설명
설정 소스 전환	[아리아 서버] / [실제 서버] 탭
Host 속성 폼	name, appBase, unpackWARs, autoDeploy, xmlBase 등
수정 / 저장 / 취소	Host 설정 편집 제어
가상 호스트 삭제	선택 vhost 제거
사이드바 + 아이콘	가상 호스트 제목 오른쪽 — 새 vhost 추가 모달 열기

2. 가상 호스트 추가

에이전트 사이드바 **가상 호스트** 제목 오른쪽 + 아이콘을 클릭하면 **가상 호스트 추가** 모달이 열립니다.



모달 입력



항목	필수	설명
name	✓	Host 이름. localhost 또는 영문·숫자·.·- 조합의 도메인 형식 (예: www.example.com)
appBase	✓	웹 애플리케이션 기준 디렉터리 (기본값 webapps)
unpackWARs		WAR 자동 압축 해제 여부 — 체크 시 true , 미체크 시 false
autoDeploy		자동 배포 여부 — 체크 시 true , 미체크 시 false

각 항목 오른쪽 **기본값 확인** 버튼으로 Tomcat 기본값을 확인할 수 있습니다.

추가 절차

1. 사이드바 **가상 호스트** 옆 **+** 아이콘을 클릭합니다.
2. 모달에서 **설정 항목**을 참고해 name , appBase 등을 입력합니다.
3. name 은 비울 수 없으며, 형식이 맞지 않으면 입력란 아래 경고가 표시됩니다.
4. appBase 도 필수입니다.
5. **완료**를 클릭합니다.
6. {name} Host 를 추가 하시겠습니까? 확인 창에서 **확인**을 누르면 아리아 작업본에 Host가 추가되고, 사이드바 vhost 목록에 표시됩니다.
7. **설정 변경 반영** 절차에 따라 Tomcat에 적용합니다.

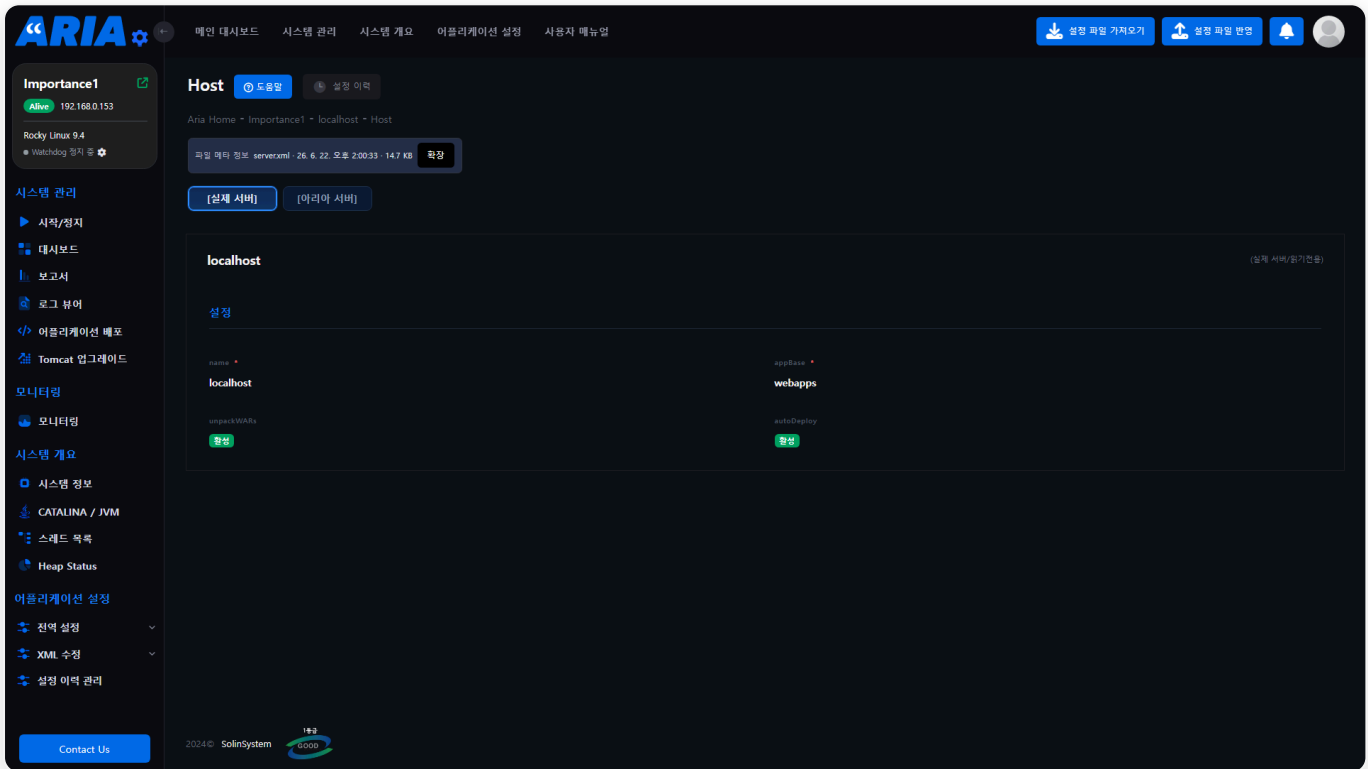
버튼	동작
완료	입력값 검증 후 추가 확인 → 아리아 작업본에 Host 등록
취소	모달을 닫고 입력을 취소

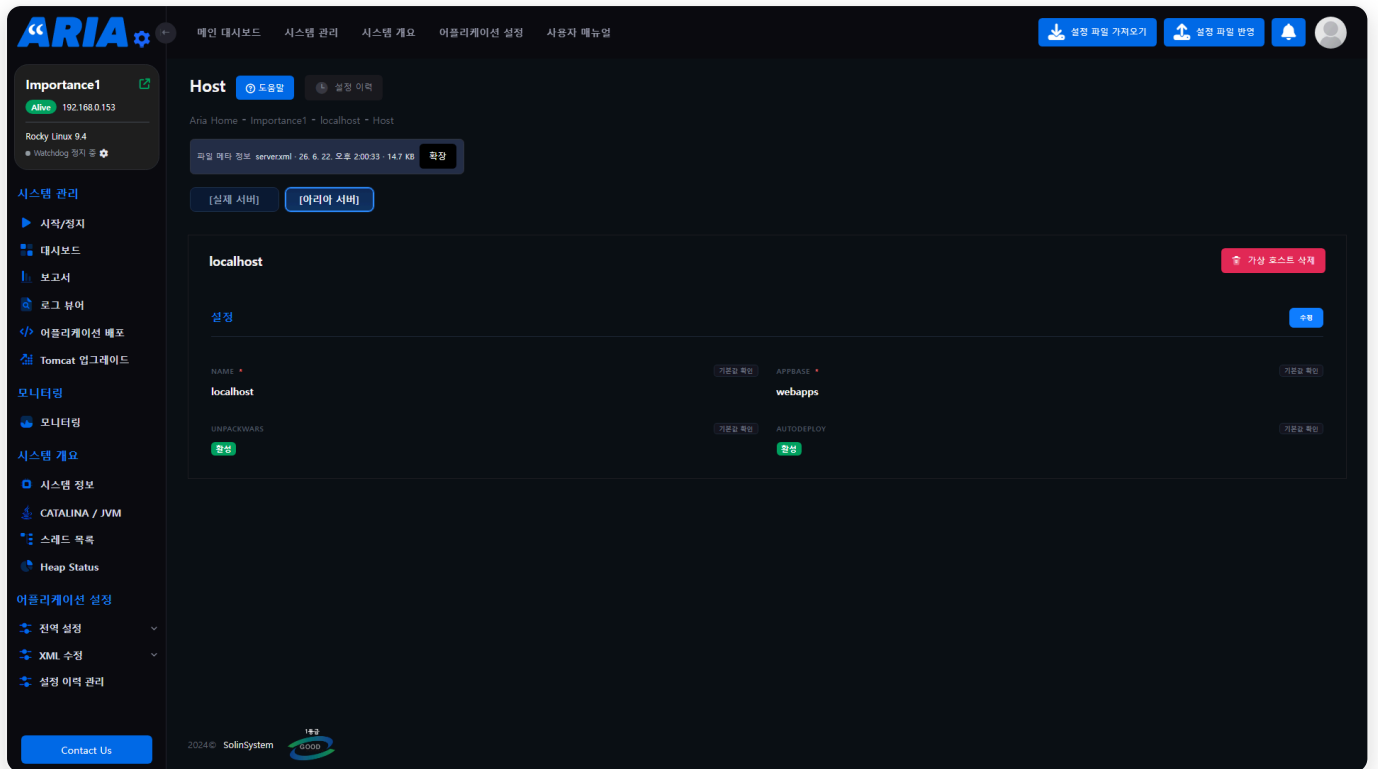
참고: 동일한 `name` 이 이미 있으면 **동일한 호스트명이 존재합니다.** 오류가 표시됩니다.

6.21.3 3. 설정 소스 전환

[아리아 서버] 탭은 선택한 vhost의 Host 속성을 ARIA 작업본에서 편집하는 화면입니다. **저장** 시 반영 대기 목록에 등록됩니다.

[실제 서버] 탭은 Tomcat에 적용된 Host 설정을 **읽기 전용**으로 조회합니다.



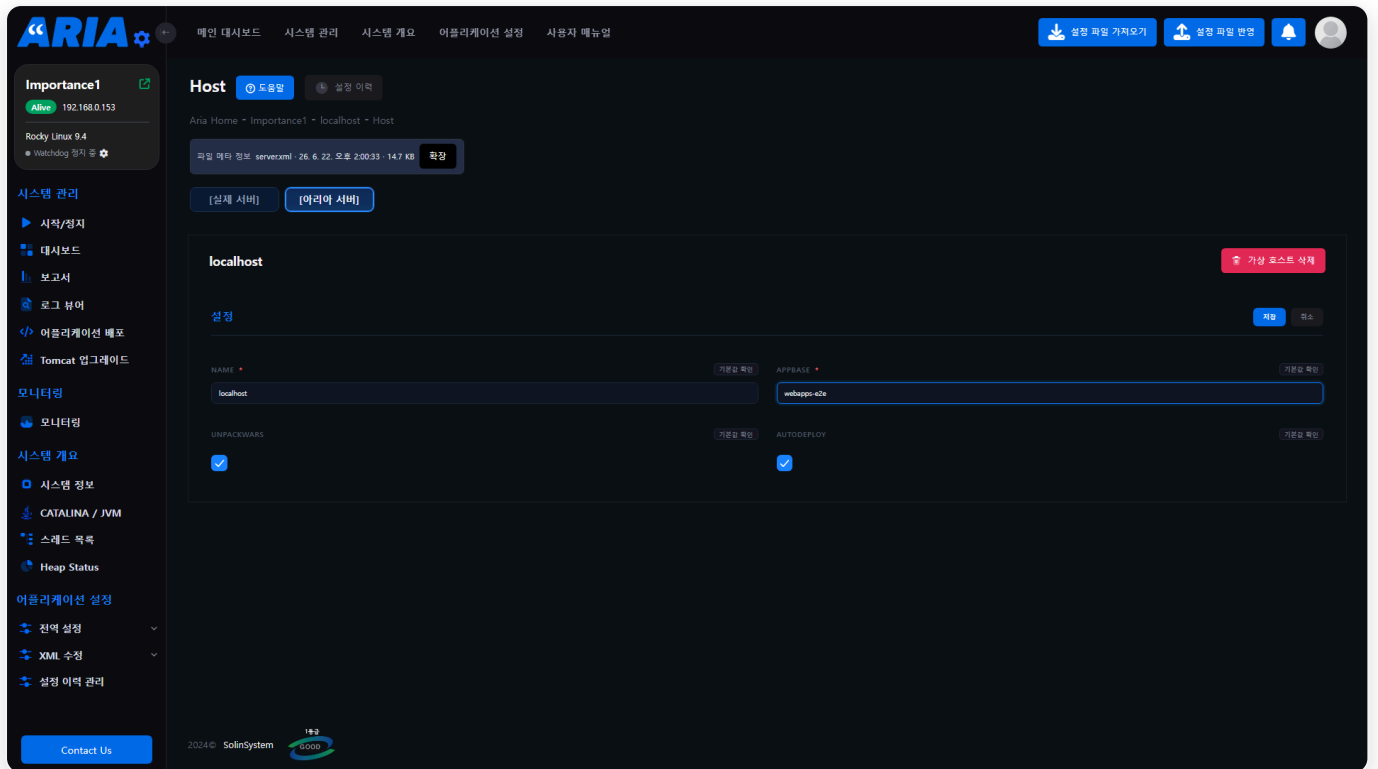


6.21.4 4. 설정 항목

항목	설명	설정 가이드
name	server.xml 의 <Host> 요소 이름입니다. HTTP 요청의 Host 헤더와 매칭되어 어느 vhost로 라우팅할지 결정합니다.	필수. DNS·요청 Host 헤더와 일치 (예: localhost, www.example.com)
appBase	이 Host에 배포되는 웹 애플리케이션(WAR·폴더)의 기준 디렉터리입니다.	필수. Tomcat 기준 경로 (예: webapps)
unpackWARs	appBase 에 배치된 WAR 파일을 자동으로 압축 해제할지 여부입니다.	true : 배포 시 WAR 풀기. false : WAR 그대로 실행
autoDeploy	appBase 에 새 WAR·폴더가 추가되면 자동으로 배포할지 여부입니다.	개발: true. 운영: 보안·안정성을 위해 false 검토

6.21.5 5. Host 설정 편집

1. [아리아 서버] 탭에서 **수정** 버튼을 클릭합니다.

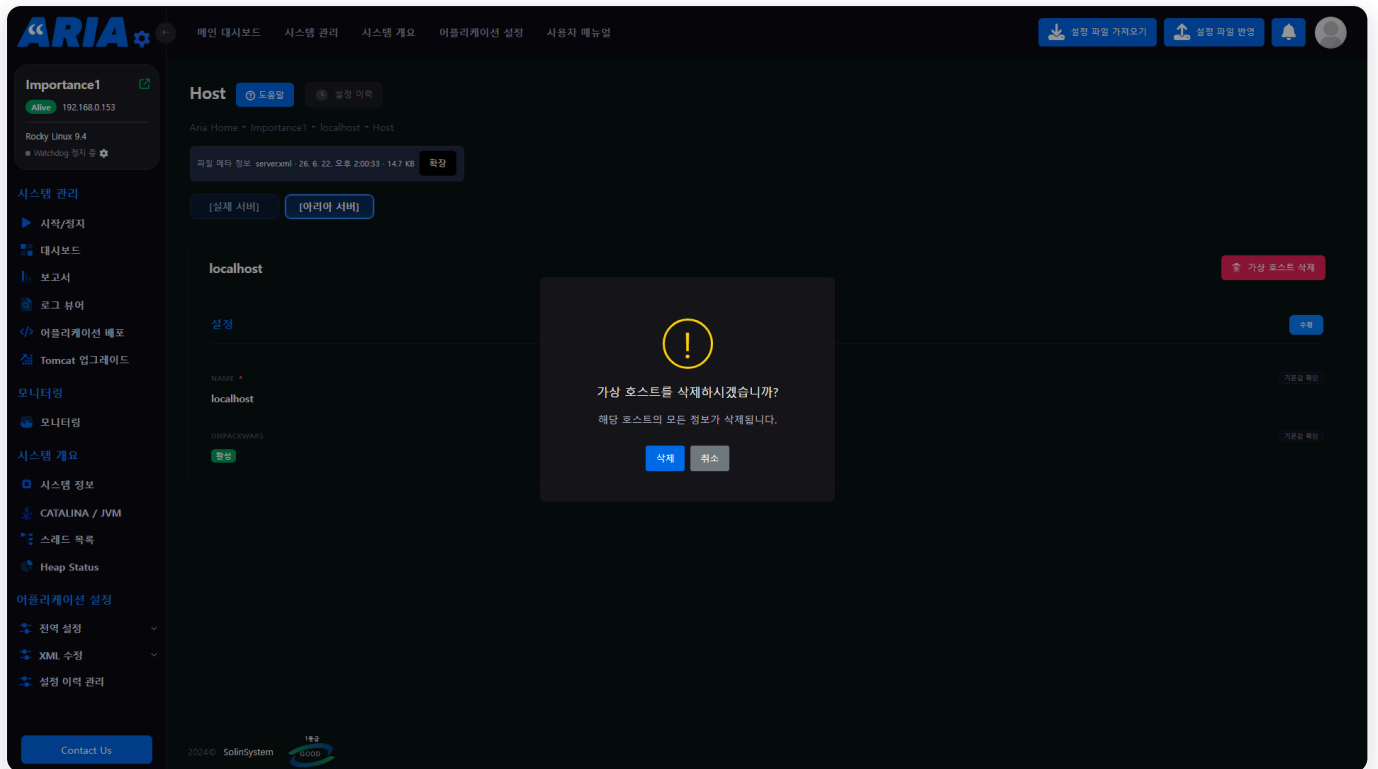


1. **설정 항목**을 참고해 Host 속성을 변경합니다.
2. **저장**을 클릭하면 아리아 작업본에 반영되고 반영 대기 목록에 등록됩니다.

버튼	동작
수정	Host 편집 모드 진입
저장	아리아 작업본에 저장 (반영 대기)
취소	편집 중 변경 사항 되돌림

가상 호스트 삭제

1. **가상 호스트 삭제** 버튼을 클릭합니다.
2. 확인 모달에서 삭제를 확정합니다.



주의: 삭제는 `server.xml` 의 해당 Host 정의를 제거합니다. 반영 전까지는 아리아 작업본에만 반영됩니다.

6.21.6 6. 서버 반영

설정 변경 반영 절차에 따라 적용하세요.

6.21.7 💡 활용 팁

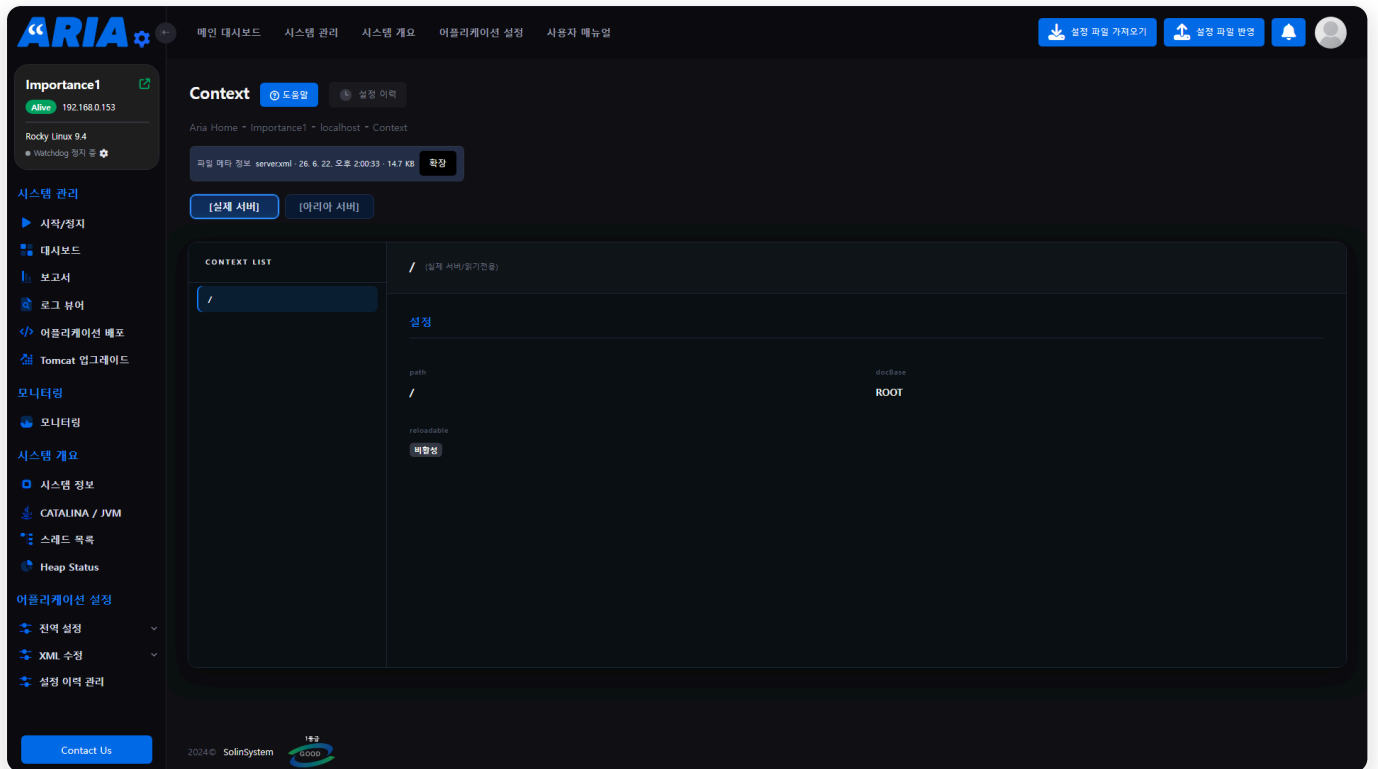
- **appBase 경로:** 변경 시 **어플리케이션 배포** 목록과 실제 WAR 배포 경로를 함께 확인하세요.
- **Context·로그:** 같은 vhost의 **가상 호스트 - Context**, **가상 호스트 - Access Log**도 함께 점검하세요.

6.22 가상 호스트 - Context

경로: `/servers/management/config/vhost/context?agent_id={agent_id}&vhost={vhost}`

선택한 가상 호스트에 속한 **Context** 목록과 상세 설정을 웹 UI로 편집하는 페이지입니다. `path`, `docBase`, `reloadable` 등 Context 속성을 관리합니다.

6.22.1 화면

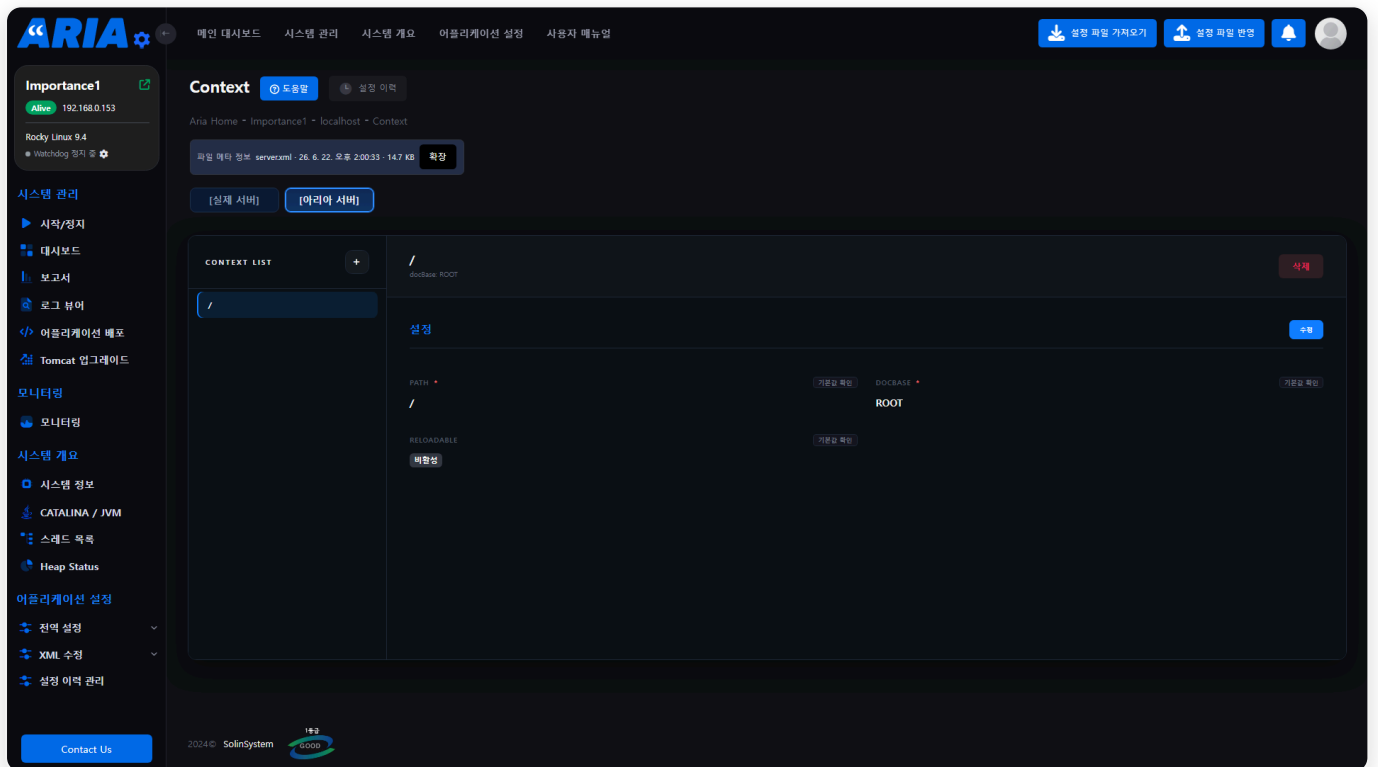
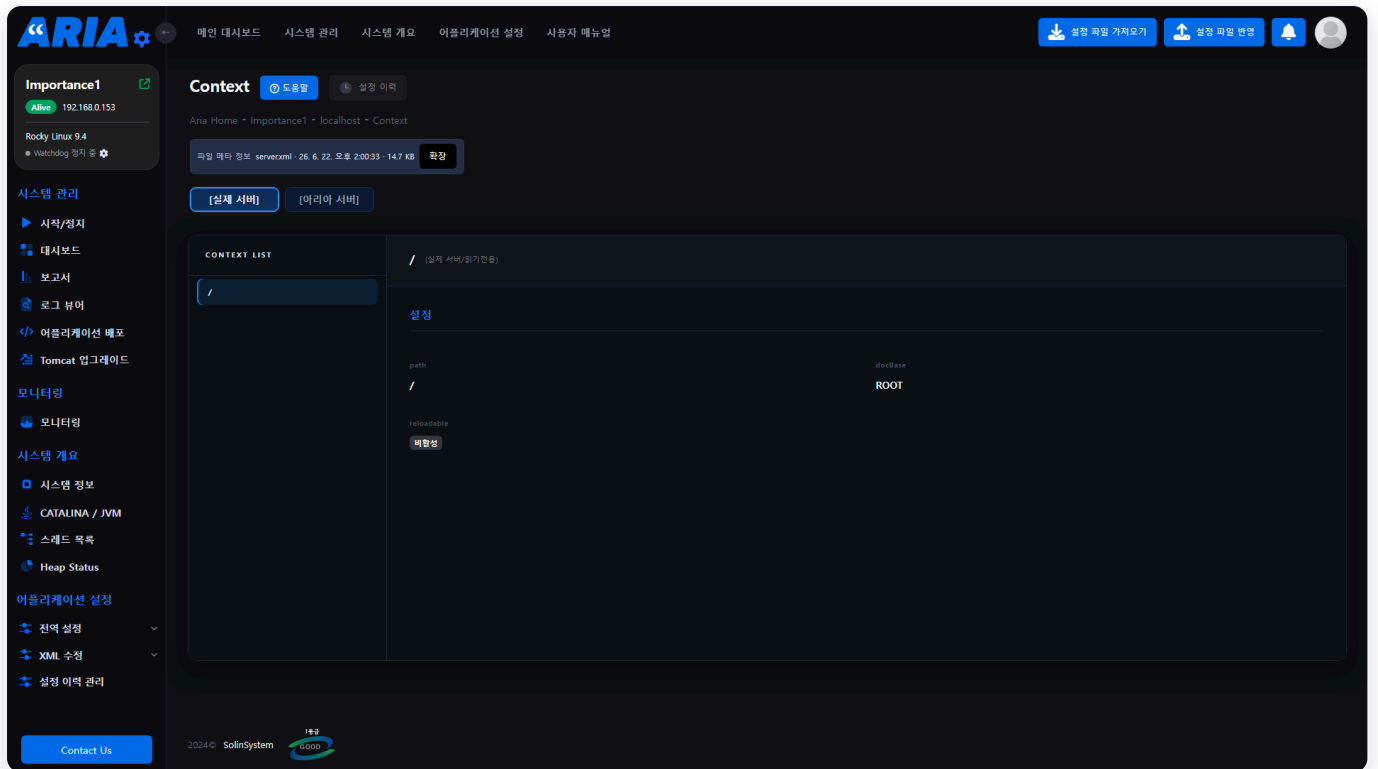


6.22.2 1. 화면 구성

영역	설명
설정 소스 전환	[아리아 서버] / [실제 서버] 탭
Context 목록	vhost 내 Context 폴더형 목록
Context 상세	path, docBase, reloadable 등
수정 / 저장 / 취소	선택 Context 편집 제어

2. 설정 소스 전환

[아리아 서버] 탭은 선택한 vhost의 Context를 ARIA 작업본에서 편집하는 화면입니다. **저장** 시 반영 대기 목록에 등록됩니다.
[실제 서버] 탭은 Tomcat에 적용된 Context 설정을 읽기 전용으로 조회합니다.

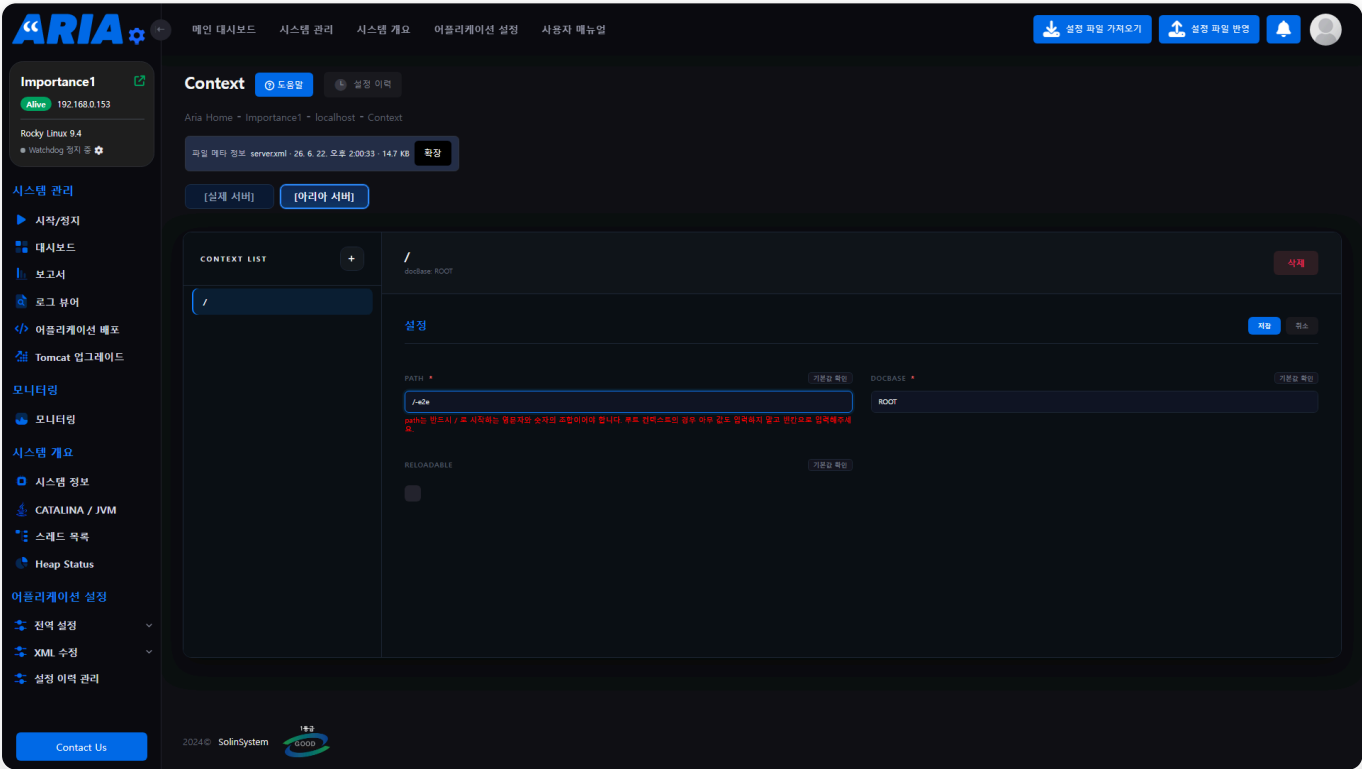


6.22.3 3. 설정 항목

항목	설명	설정 가이드
path	Context의 URL 컨텍스트 경로입니다. 이 경로로 들어온 요청이 해당 docBase 애플리케이션으로 전달됩니다.	필수. 웹 접근 경로 (예: / , /myapp). 빈 문자열은 루트
docBase	Context가 서비스하는 웹 애플리케이션의 실제 위치입니다. WAR 파일·디렉터리·절대 경로를 지정할 수 있습니다.	필수. Host appBase 기준 상대 경로 또는 절대 경로
reloadable	클래스-리소스 변경 시 Context를 자동 리로드할지 여부입니다. 개발 편의와 운영 안정성 사이의 트레이드오프가 있습니다.	개발: true . 운영: false 권장 — 메모리 누수·성능 영향 주의

6.22.4 4. Context 설정 편집

1. [아리아 서버] 탭에서 Context를 선택하고 수정을 클릭합니다.



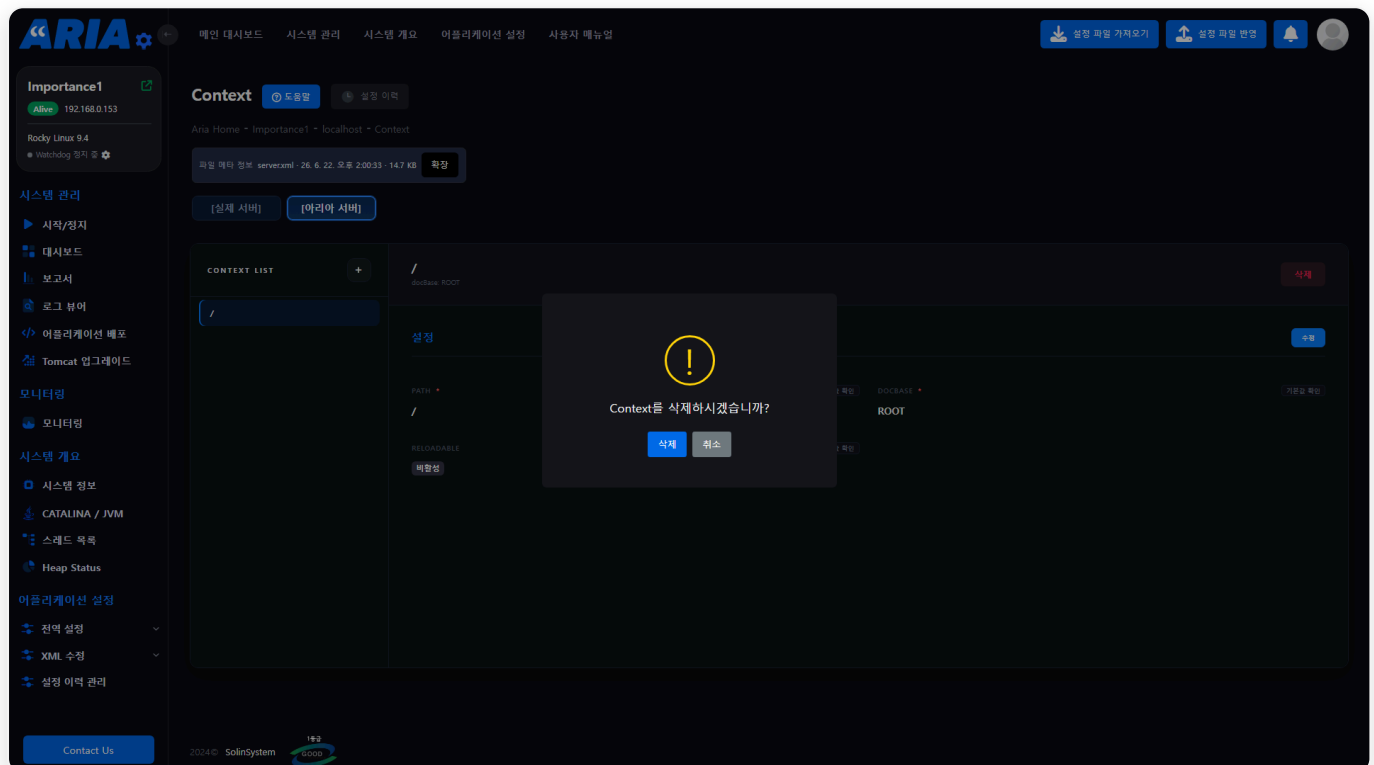
1. **설정 항목**을 참고해 Context 속성을 변경합니다.
2. **저장**을 클릭하면 아리아 작업본에 반영되고 반영 대기 목록에 등록됩니다.

버튼	동작
수정	선택 Context 편집 모드 진입

버튼	동작
저장	아리아 작업본에 저장 (반영 대기)
취소	편집 중 변경 사항 되돌림

Context 삭제

1. 삭제할 Context를 선택하고 **삭제** 버튼을 클릭합니다.
2. 확인 모달에서 삭제를 확정합니다.



6.22.5 5. 서버 반영

설정 변경 반영 절차에 따라 적용하세요.

6.22.6 💡 활용 팁

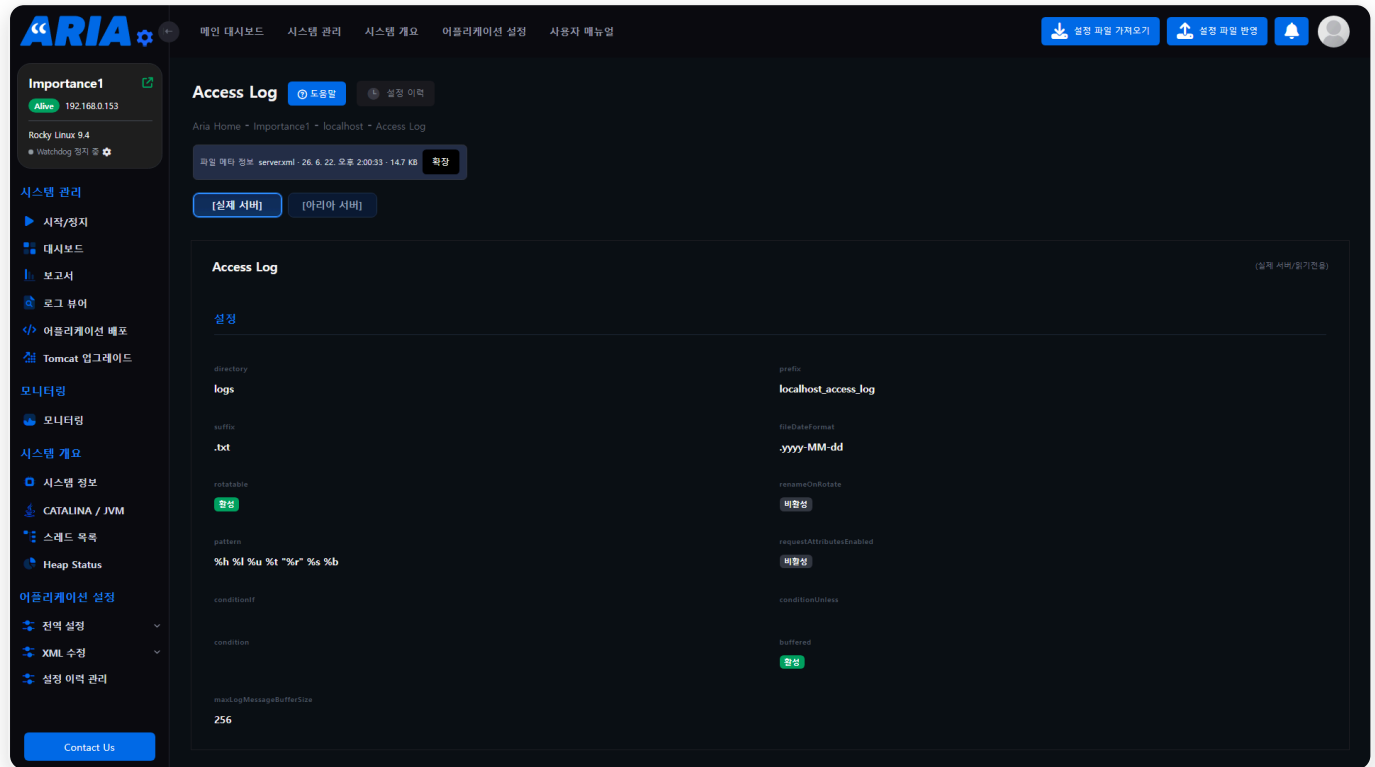
- **배포와 연계:** WAR 배포-리로드는 **어플리케이션 배포**에서 제어할 수 있습니다.
- **전역 context.xml:** Host 공통 설정은 **context.xml**도 참고하세요.

6.23 가상 호스트 - Access Log

경로: /servers/management/config/vhost/accesslog?agent_id={agent_id}&vhost={vhost}

선택한 가상 호스트의 **Access Log Valve** 설정(directory, prefix, suffix, pattern 등)을 웹 UI로 편집하는 페이지입니다.

6.23.1 화면



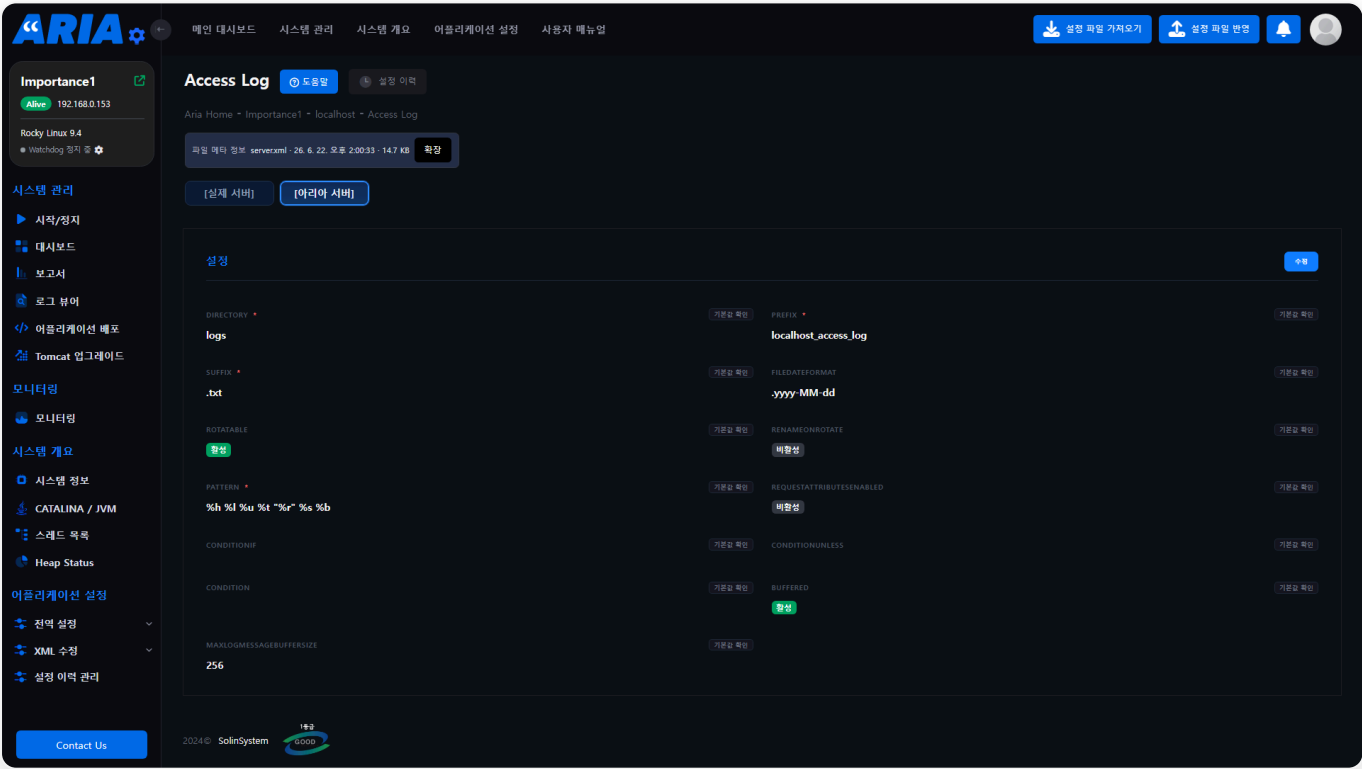
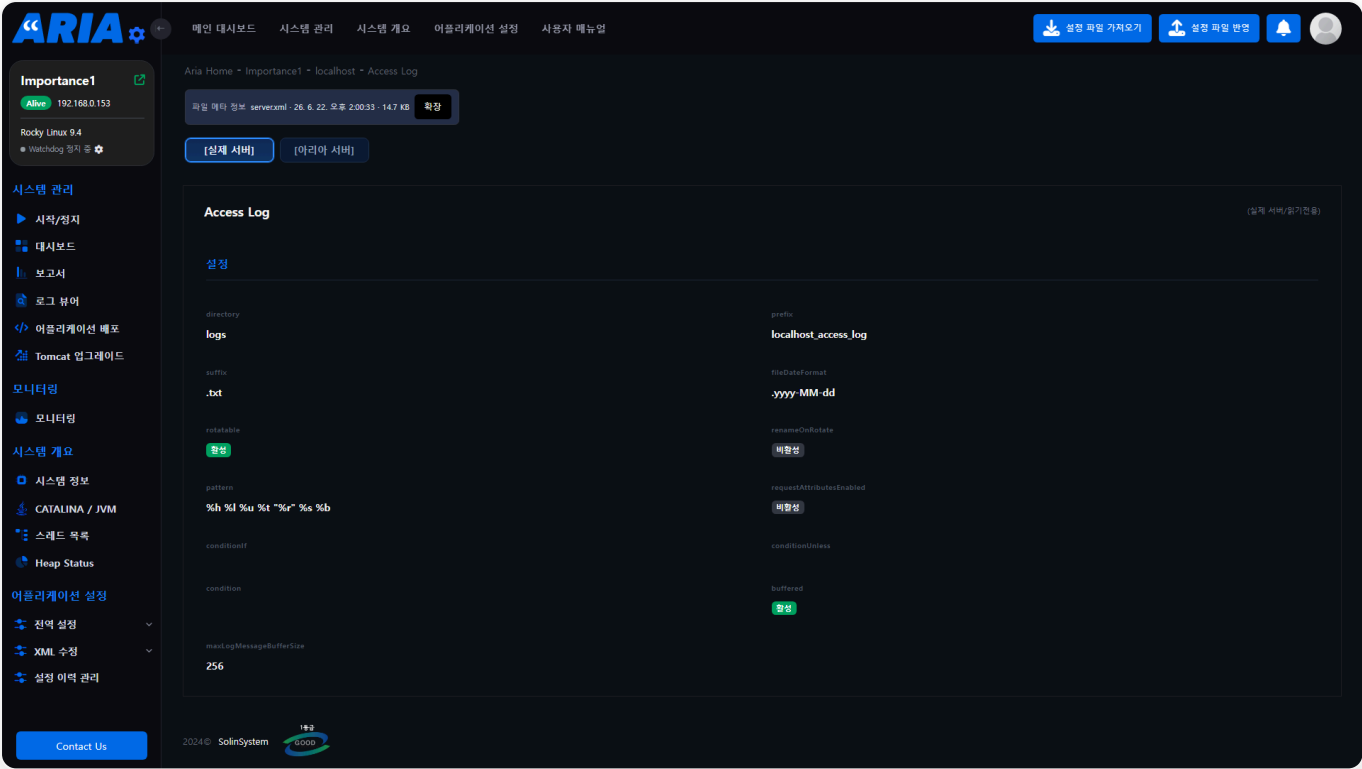
6.23.2 1. 화면 구성

영역	설명
설정 소스 전환	[아리아 서버] / [실제 서버] 탭
Access Log 폼	directory, prefix, suffix, pattern, rotatable 등
Pattern Format	Access Log 패턴 형식 참조 모달
수정 / 저장 / 취소	Access Log 설정 편집 제어

2. 설정 소스 전환

[아리아 서버] 탭은 선택한 vhost의 Access Log Valve를 ARIA 작업본에서 편집하는 화면입니다. **저장** 시 반영 대기 목록에 등록됩니다.

[실제 서버] 탭은 Tomcat에 적용된 Access Log 설정을 읽기 전용으로 조회합니다.



6.23.3 3. 설정 항목

항목	설명	설정 가이드
Directory	access log 파일이 생성될 디렉터리의 절대·상대 경로입니다. 상대 경로는 <code>\${catalina.base}</code> 기준입니다.	필수. 예: <code>logs</code> . Tomcat 프로세스 쓰기 권한 확인
prefix	로그 파일 이름 앞에 붙는 접두사입니다.	필수. 기본 <code>access_log</code> . 예: <code>localhost_access_log</code>
suffix	로그 파일 이름 끝에 붙는 접미사입니다.	필수. 예: <code>.txt</code> , <code>.log</code> . 기본은 빈 문자열
fileDateFormat	로그 파일 이름에 포함할 날짜 형식입니다. 로테이션 시 파일명에 타임스탬프를 반영합니다.	<code>yyyy-MM-dd</code> 등. <code>rotatable=true</code> 시 사용
rotatable	일별(또는 주기별) 로그 로테이션 여부입니다.	기본 <code>true</code> 권장 — 디스크 관리 용이
renameOnRotate	<code>true</code> 면 현재 로그 파일명에는 날짜를 넣지 않고, 로테이션으로 생성되는 파일에만 <code>fileDateFormat</code> 을 적용합니다.	기본 <code>false</code>
pattern	access log 한 줄에 기록될 필드 형식입니다. <code>%h</code> (클라이언트 IP), <code>%t</code> (시각), <code>%r</code> (요청) 등 토큰 조합입니다.	필수. 로그 뷰어 Access Log 상세 분석 을 사용하려면 아래 지원 pattern 중 하나로 설정하세요. 그 외 일반 조회만 필요하면 Common <code>%h %l %u %t "%r" %s %b</code> 등도 가능합니다. Pattern Format 모달 참고
requestAttributesEnabled	Request 속성으로 원격 주소·호스트·포트·프로토콜을 재정의했는지 확인할지 여부입니다.	기본 <code>false</code>
conditionIf	조건부 로깅입니다. <code>ServletRequest.getAttribute(값) != null</code> 인 요청만 기록합니다.	EL-속성명 지정
conditionUnless	조건부 로깅입니다. <code>ServletRequest.getAttribute(값) == null</code> 인 요청만 기록합니다.	EL-속성명 지정

항목	설명	설정 가이드
condition	<code>conditionUnless</code> 와 동일합니다. 이전 버전 호환용입니다.	이전 설정과의 호환 시에만 사용
buffered	로그 기록 시 버퍼링 여부입니다. <code>false</code> 면 요청마다 즉시 디스크에 기록합니다.	기본 <code>true</code> . 즉시 기록이 필요하면 <code>false</code>
maxLogMessageBufferSize	로그 메시지 버퍼 크기입니다.	기본 256. <code>buffered=true</code> 시 조정

Access Log 상세 분석용 pattern

로그 뷰어의 **Access Log 상세 분석**(트래픽 추이, 응답 코드·URI별 성능 분석 등)을 사용하려면, 이 화면의 `pattern` 을 ARIA가 지원하는 형식 **둘 중 하나로** 맞춰야 합니다. 패턴이 다르면 분석 실행 시 **포맷 불일치 경고**가 표시됩니다.

지원 pattern 및 필드 설명은 [로그 뷰어 §6-1 지원 로그 포맷](#)을 참고하세요.

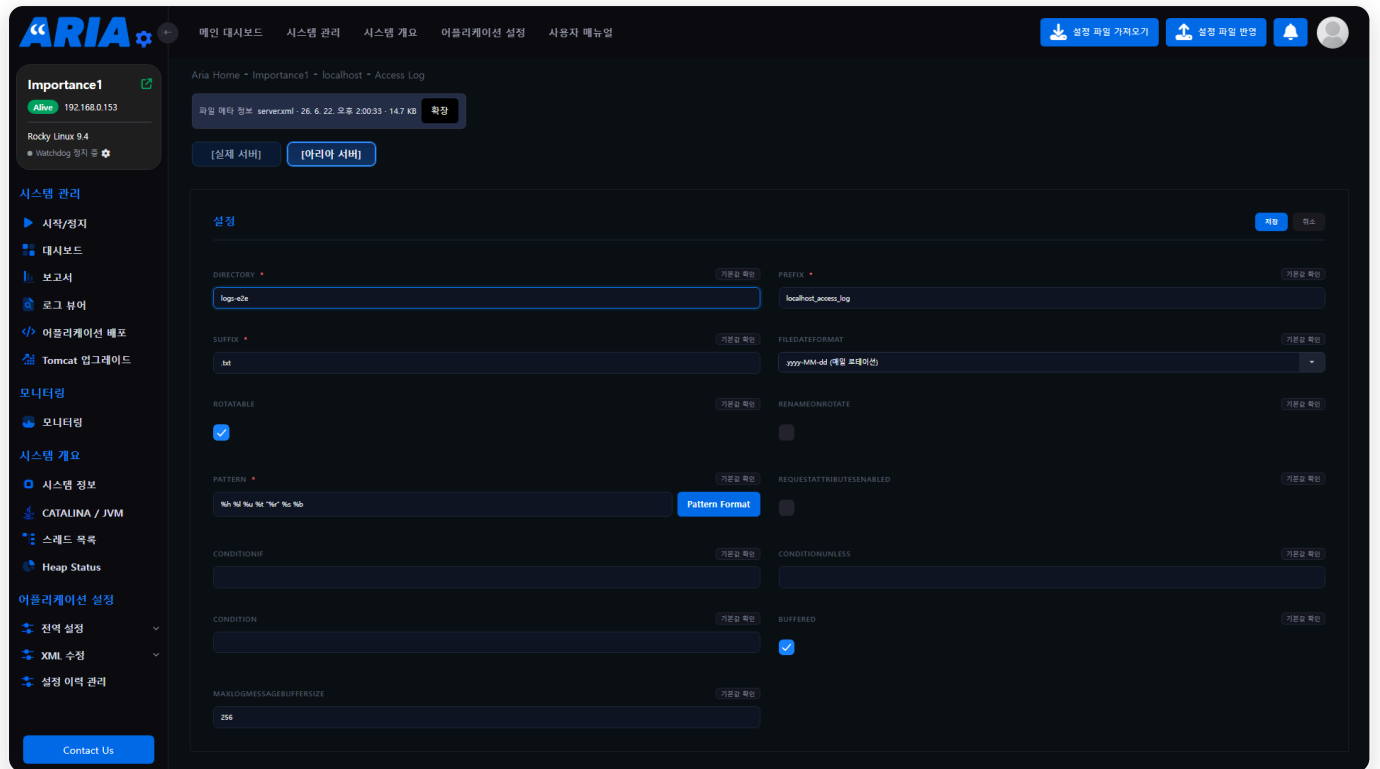
구분	설정 pattern
표준 Combined (처리 시간 포함)	<code>%h %l %u %t "%r" %s %b %D</code>
확장 헤더 (Referer·User-Agent, 접두사)	<code>%h %l %u %t %r %{Referer}i %{User-Agent}i length-%b status-%s dur-%D</code>

- 표준 패턴 예시 로그: 127.0.0.1 - - [19/Sep/2025:09:56:02 +0900] "GET /index.html HTTP/1.1" 200 1204 45
- 확장 패턴 예시 로그: 127.0.0.1 - - [19/Sep/2025:09:56:02 +0900] GET /index.html HTTP/1.1 http://localhost/Mozilla/5.0 length-1204 status-200 dur-45

`pattern` 저장 후 **설정 변경 반영**으로 Tomcat에 적용한 뒤, [로그 뷰어](#) `localhost_access_log` 탭에서 상세 분석을 실행해 보세요.

6.23.4 4. Access Log 설정 편집

- [아리아 서버] 탭에서 **수정** 버튼을 클릭합니다.



1. **설정 항목**을 참고해 로그 디렉터리·파일명·pattern 등을 변경합니다.
2. 패턴 형식이 필요하면 **Pattern Format** 버튼으로 참조 모달을 엽니다.
3. **저장**을 클릭하면 아리아 작업본에 반영되고 반영 대기 목록에 등록됩니다.

버튼	동작
수정	Access Log 편집 모드 진입
Pattern Format	%h %l %u %t 등 패턴 토큰 참조 모달
저장	아리아 작업본에 저장 (반영 대기)
취소	편집 중 변경 사항 되돌림

6.23.5 5. 서버 반영

설정 변경 반영 절차에 따라 적용하세요.

6.23.6 💡 활용 팁

- **상세 분석 연동**: 트래픽·URI 성능 분석이 필요하면 pattern 을 지원 pattern으로 맞춘 뒤 로그 뷰어 Access Log 상세 분석을 사용하세요.

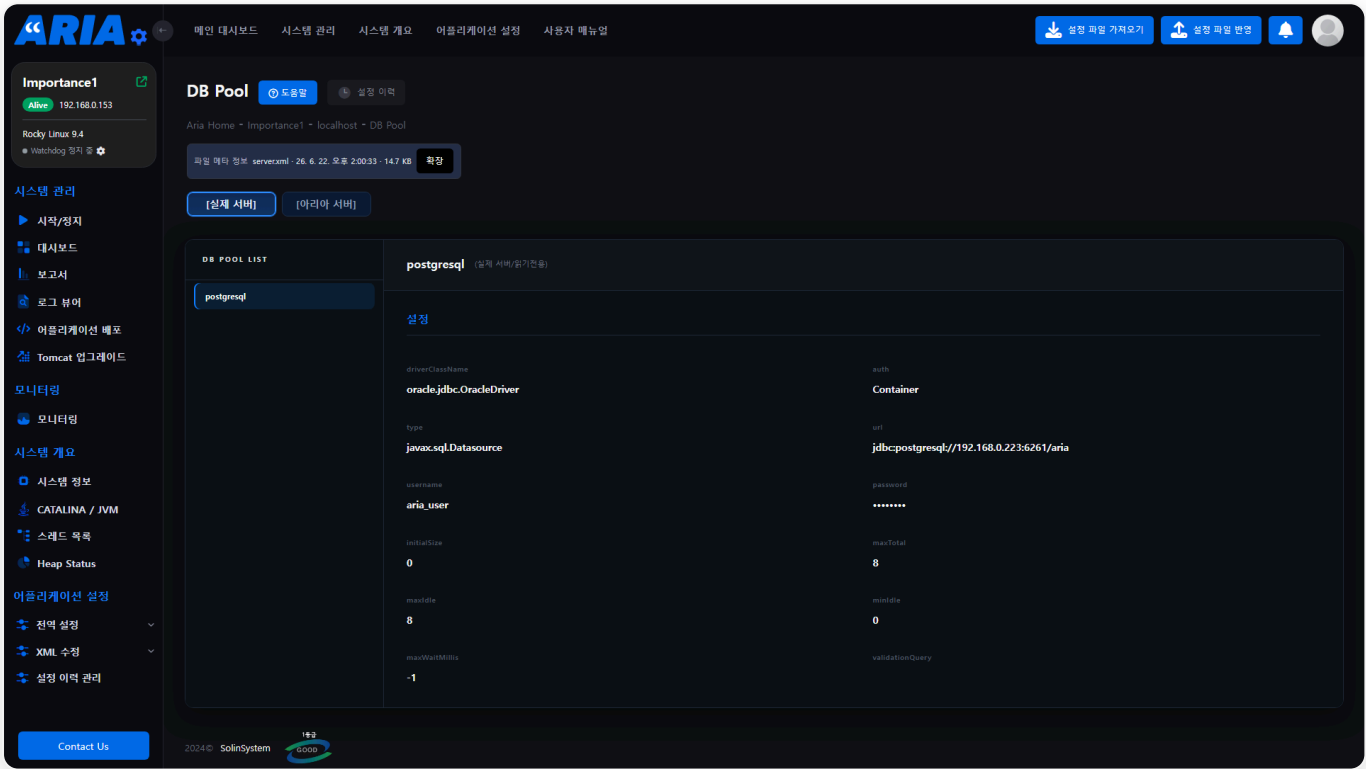
- **로그 확인:** 반영 후 [로그 뷰어](#)에서 access log 파일을 확인하세요.
- **패턴 검증:** `pattern` 변경 후 실제 로그 한 줄을 열어 필드 구분이 의도대로인지 확인하세요.

6.24 가상 호스트 - DBPool

경로: `/servers/management/config/vhost/dbpool?agent_id={agent_id}&vhost={vhost}`

선택한 가상 호스트 범위의 **JDBC Resource(DB Pool)** 설정을 웹 UI로 편집하는 페이지입니다. 해당 vhost의 Context에서 참조하는 DataSource를 관리합니다.

6.24.1 화면



6.24.2 1. 화면 구성

영역	설명
설정 소스 전환	[아리아 서버] / [실제 서버] 탭
DB Pool 목록	vhost 범위 JDBC Resource 목록
JDBC Resource 폼	<code>url</code> , 드라이버, 풀 크기 등

영역	설명
DBPool 연결 확인	저장 전 DB 연결 테스트
수정 / 저장 / 취소	편집 제어

2. 설정 소스 전환

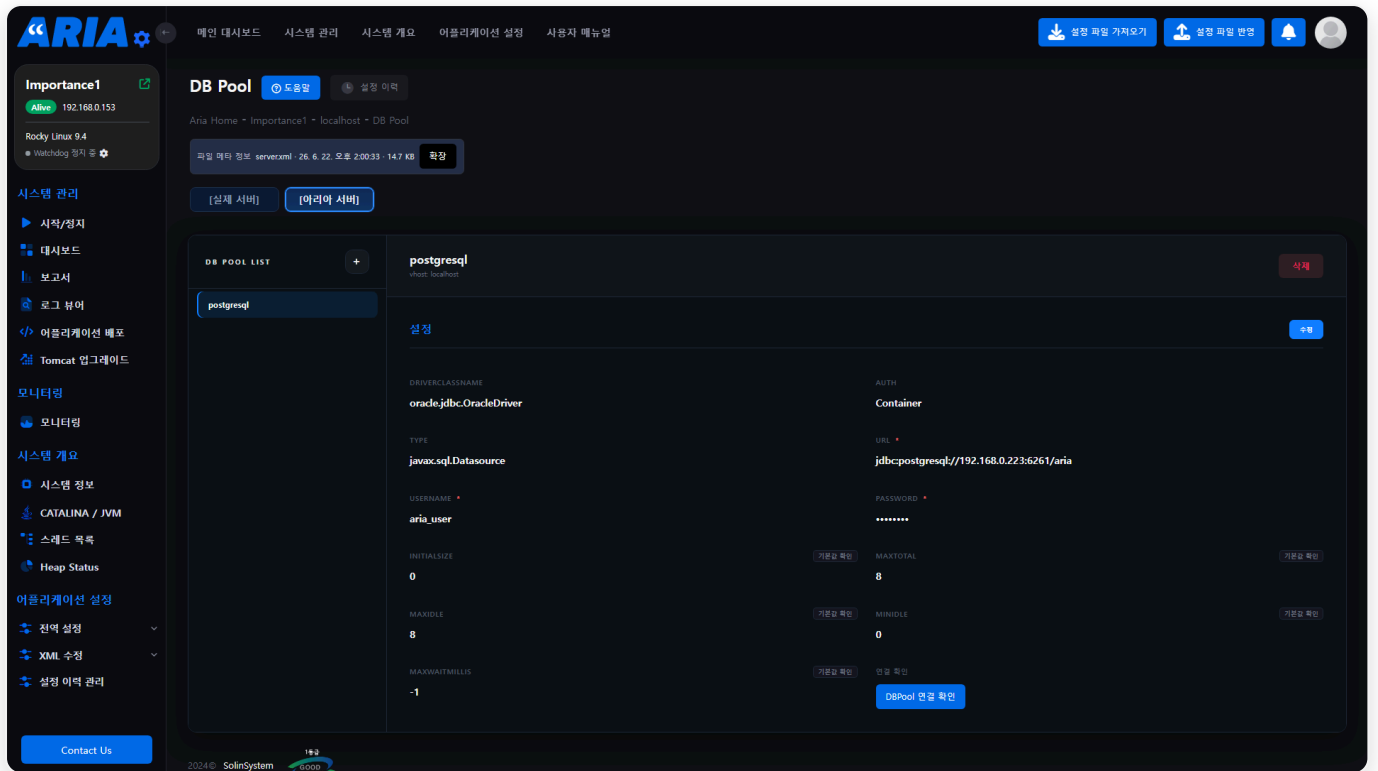
[아리아 서버] 탭은 선택한 vhost 범위의 JDBC Resource를 ARIA 작업본에서 편집하는 화면입니다. **저장** 시 반영 대기 목록에 등록됩니다.

[실제 서버] 탭은 Tomcat에 적용된 vhost DB Pool 설정을 **읽기 전용**으로 조회합니다.

The screenshot shows the ARIA DB Pool configuration page. The sidebar on the left contains navigation links for system management, monitoring, and application settings. The main content area is titled 'DB Pool' and shows a list of pools. The 'postgresql' pool is selected, and its configuration details are displayed in a table.

DB POOL LIST	
postgresql	

postgresql (실제 서버/읽기전용)	
설정	
driverClassName	auth
oracle.jdbc.OracleDriver	Container
type	url
javax.sql.DataSource	jdbc:postgresql://192.168.0.223:6261/aria
username	password
aria_user	*****
initialSize	maxTotal
0	8
maxIdle	minIdle
8	0
maxWaitMillis	validationQuery
-1	



6.24.3 3. 설정 항목

이 페이지는 **특정 가상 호스트 Host** 아래 Context에서 참조하는 DataSource만 관리합니다. 서버 전체 공유 풀은 **DB Pool 설정** (전역)에서 설정하세요.

JDBC 연결

항목	설명	설정 가이드
driverClassName	JDBC 드라이버의 클래스 이름입니다.	DBMS에 맞는 드라이버 지정 (예: <code>org.mariadb.jdbc.Driver</code>)
auth	Connection 인증 방식입니다. Tomcat 컨테이너가 계정을 관리합니다.	<code>Container</code> 고정
type	JNDI Resource 타입입니다.	<code>javax.sql.DataSource</code> 고정
url	Database 연결에 필요한 JDBC URL입니다.	필수. 호스트·포트·DB명 포함 (예: <code>jdbc:mariadb://host:3306/db</code>)
username	Database 연결에 필요한 사용자 이름입니다.	필수
password	Database 연결에 필요한 비밀번호입니다.	필수. 반영 후에도 안전하게 관리

커넥션 풀

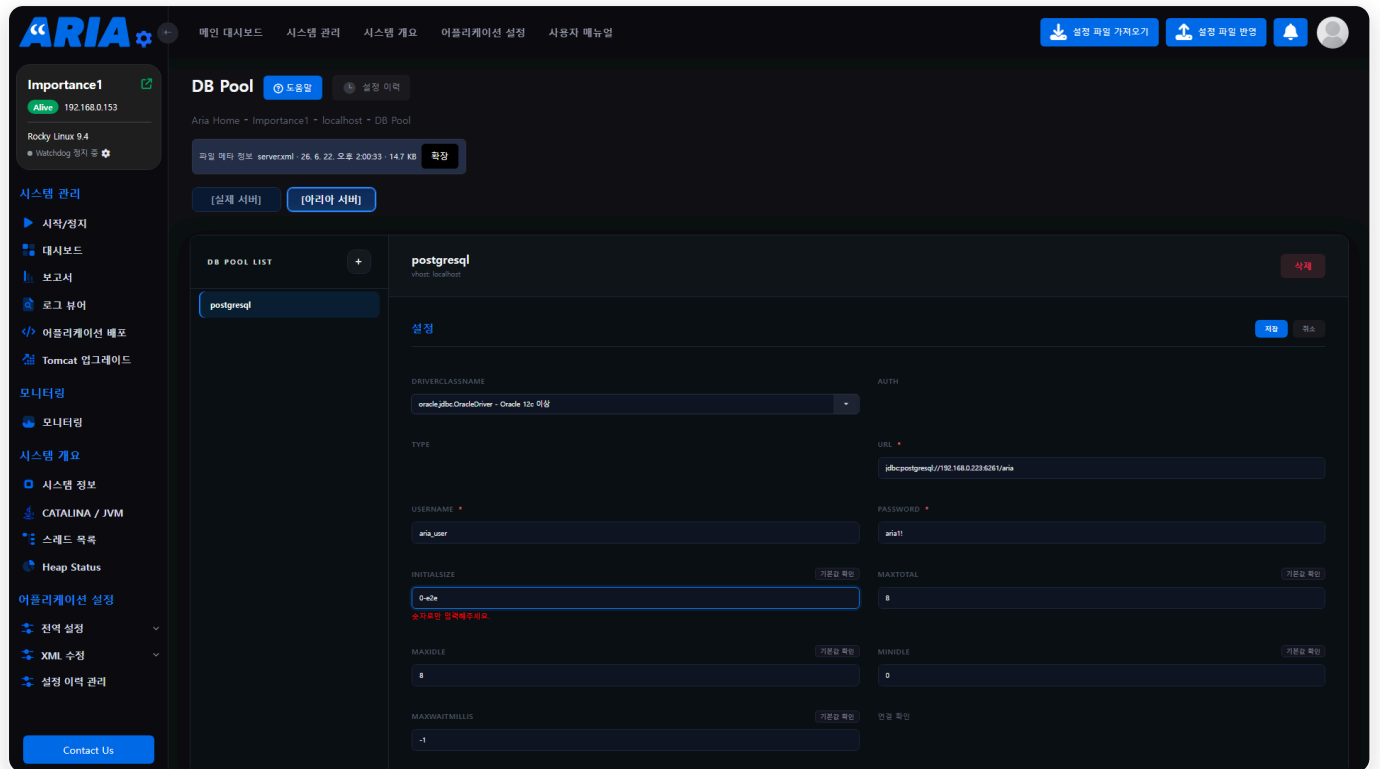
항목	설명	설정 가이드
initialSize	DB Pool 최초 구성 시 생성될 Connection 수입니다.	기본 0. 기동 시 미리 확보할 연결 수
maxTotal	DB Pool에서 사용 가능한 최대 Connection 수입니다. 음수는 무제한입니다.	기본 8. DB 서버 max_connections 이하로 설정
maxIdle	DB Pool에서 유지할 최대 유휴 Connection 수입니다. 음수는 무제한입니다.	기본 8. maxTotal 이하 권장
minIdle	최소 유휴 Connection 수입니다.	트래픽에 따라 0 ~ initialSize 수준
maxWaitMillis	사용 가능한 Connection이 없을 때 대기하는 최대 시간(ms)입니다. -1 은 무한 대기입니다.	기본 -1. 풀 고갈 시 타임아웃이 필요하면 양수 지정

연결 확인

항목	설명	설정 가이드
연결 확인	저장 전 JDBC URL·계정으로 실제 DB 접속을 테스트하는 기능입니다.	저장·반영 전 반드시 실행 권장

6.24.4 4. DB Pool 설정 편집

1. [아리아 서버] 탭에서 JDBC Resource를 선택하고 수정을 클릭합니다.

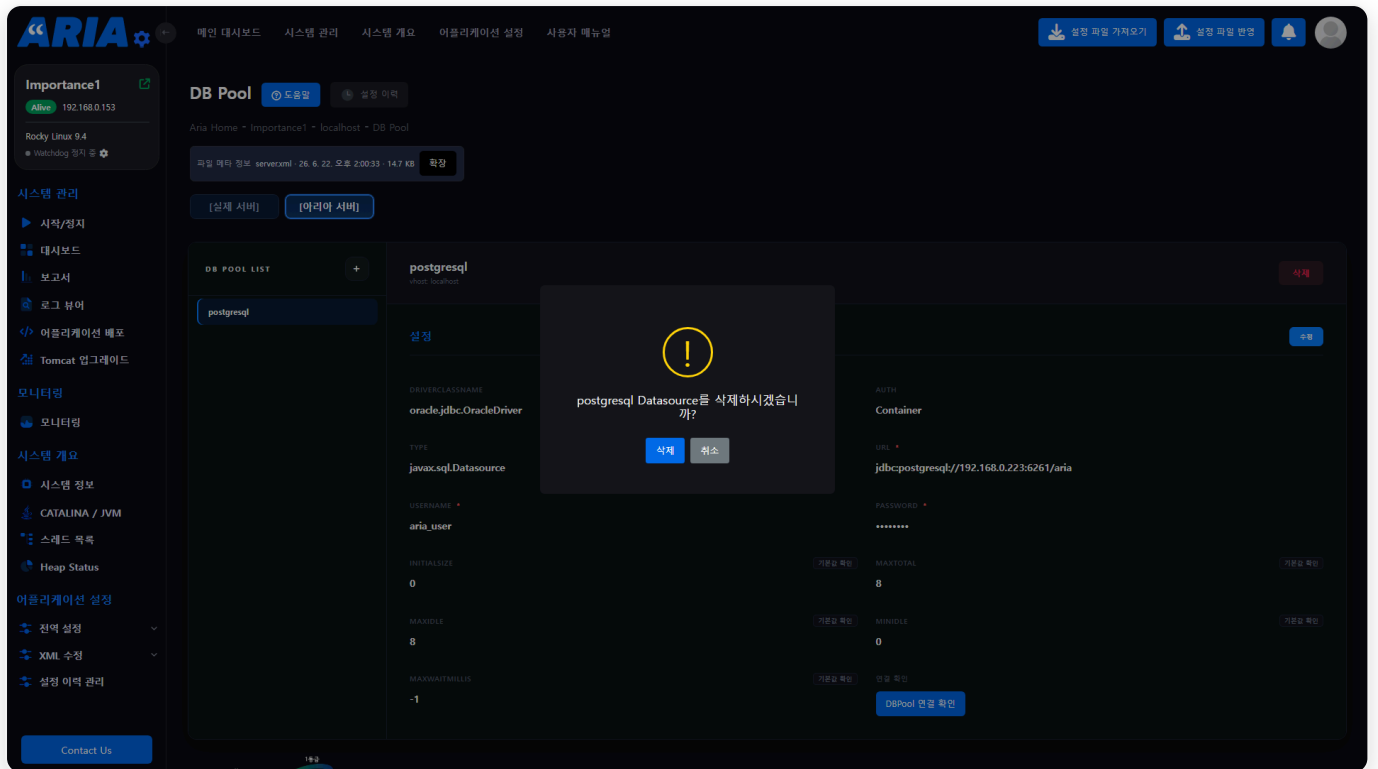


1. **설정 항목**을 참고해 URL·계정·풀 크기 등을 변경합니다.
2. **DBPool 연결 확인**으로 접속을 테스트합니다.
3. **저장**을 클릭하면 아리아 작업본에 반영되고 반영 대기 목록에 등록됩니다.

버튼	동작
수정	선택 풀 편집 모드 진입
DBPool 연결 확인	JDBC 연결 테스트
저장	아리아 작업본에 저장 (반영 대기)
취소	편집 중 변경 사항 되돌림

DB Pool 삭제

1. 삭제할 JDBC Resource를 선택하고 **삭제** 버튼을 클릭합니다.
2. 확인 모달에서 삭제를 확정합니다.



6.24.5 5. 서버 반영

설정 변경 반영 절차에 따라 적용하세요.

6.24.6 💡 활용 팁

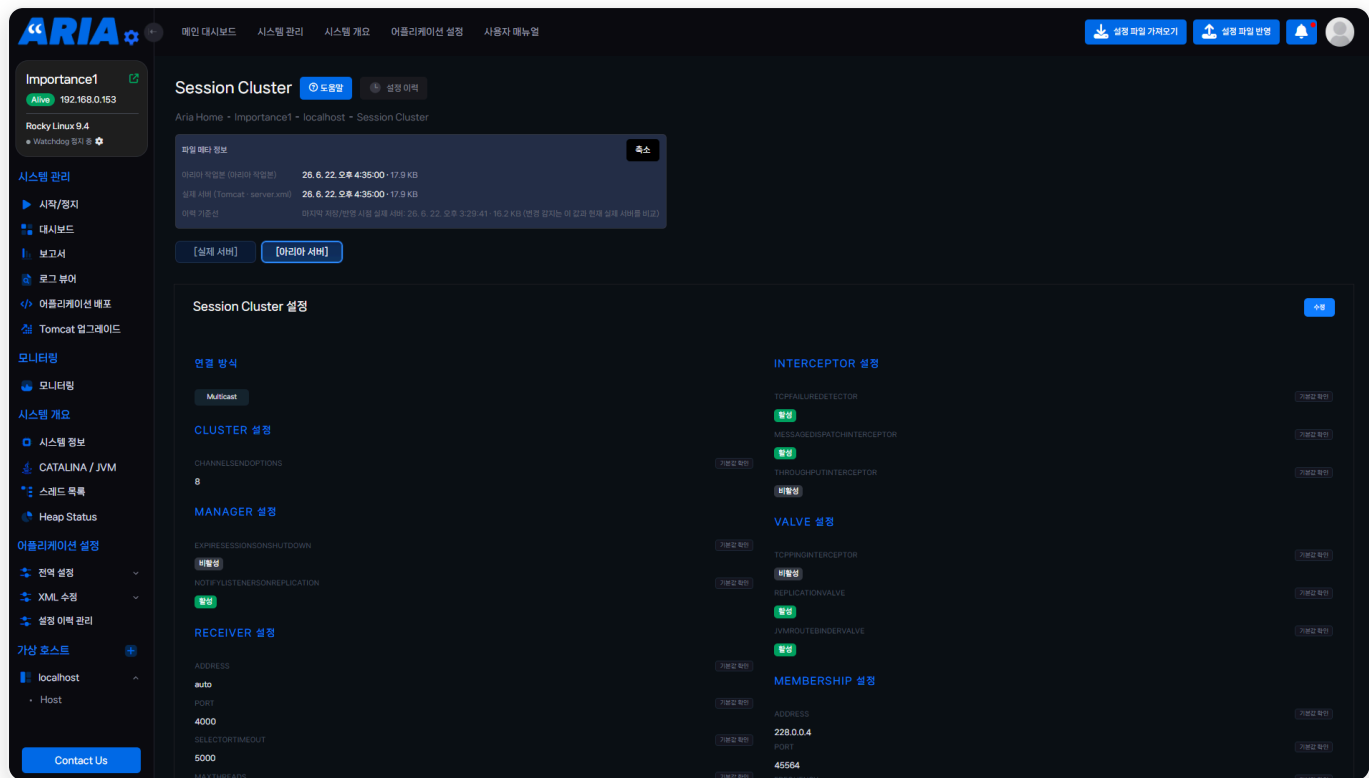
- **전역 vs vhost:** 서버 전체 공유 풀은 **DB Pool 설정**(전역)에서 관리하세요.
- **모니터링:** **모니터링**에서 DataSource 연결 수를 확인할 수 있습니다.

6.25 가상 호스트 - Cluster

경로: `/servers/management/config/vhost/cluster?agent_id={agent_id}&vhost={vhost}`

선택한 가상 호스트 범위의 **Session Cluster** 설정을 웹 UI로 편집하는 페이지입니다. 해당 Host에 속한 클러스터 Manager·Channel·Receiver·Interceptor·Valve·Membership을 관리합니다.

6.25.1 화면



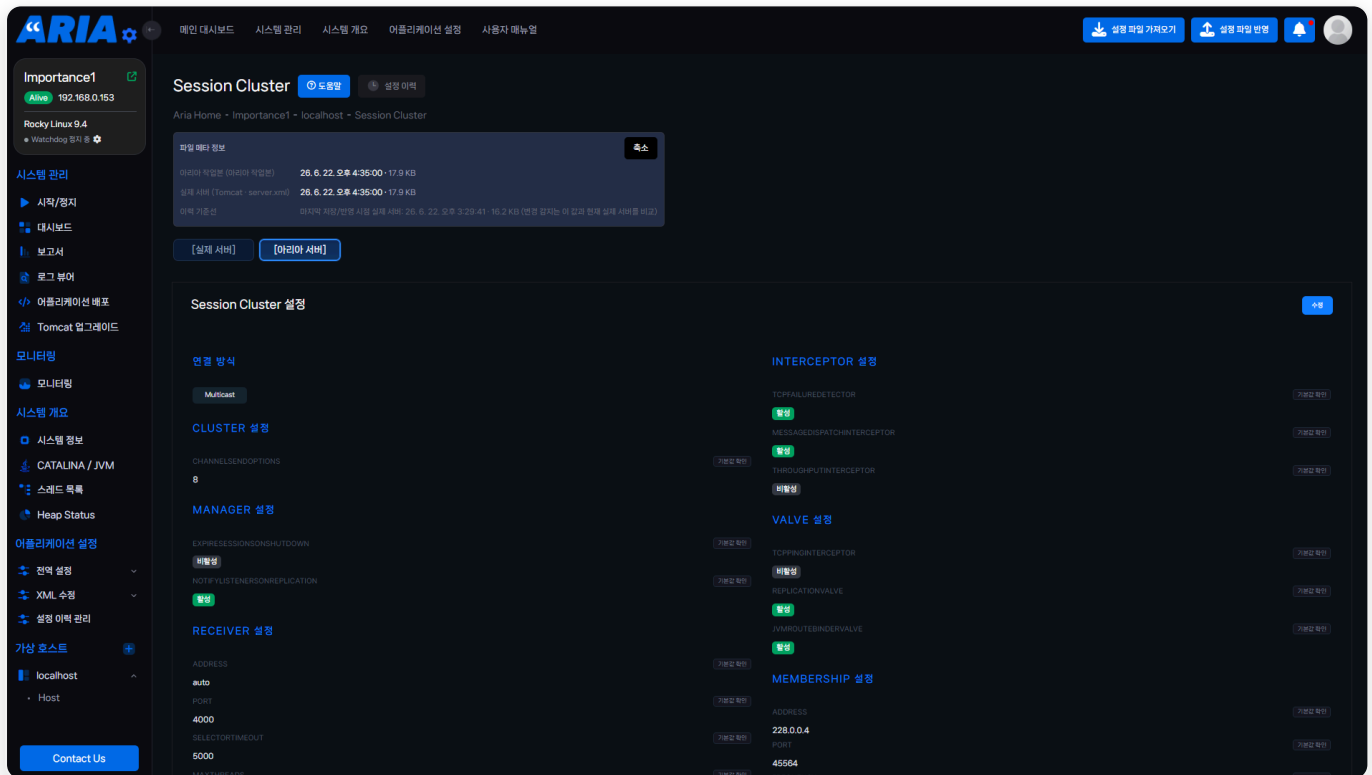
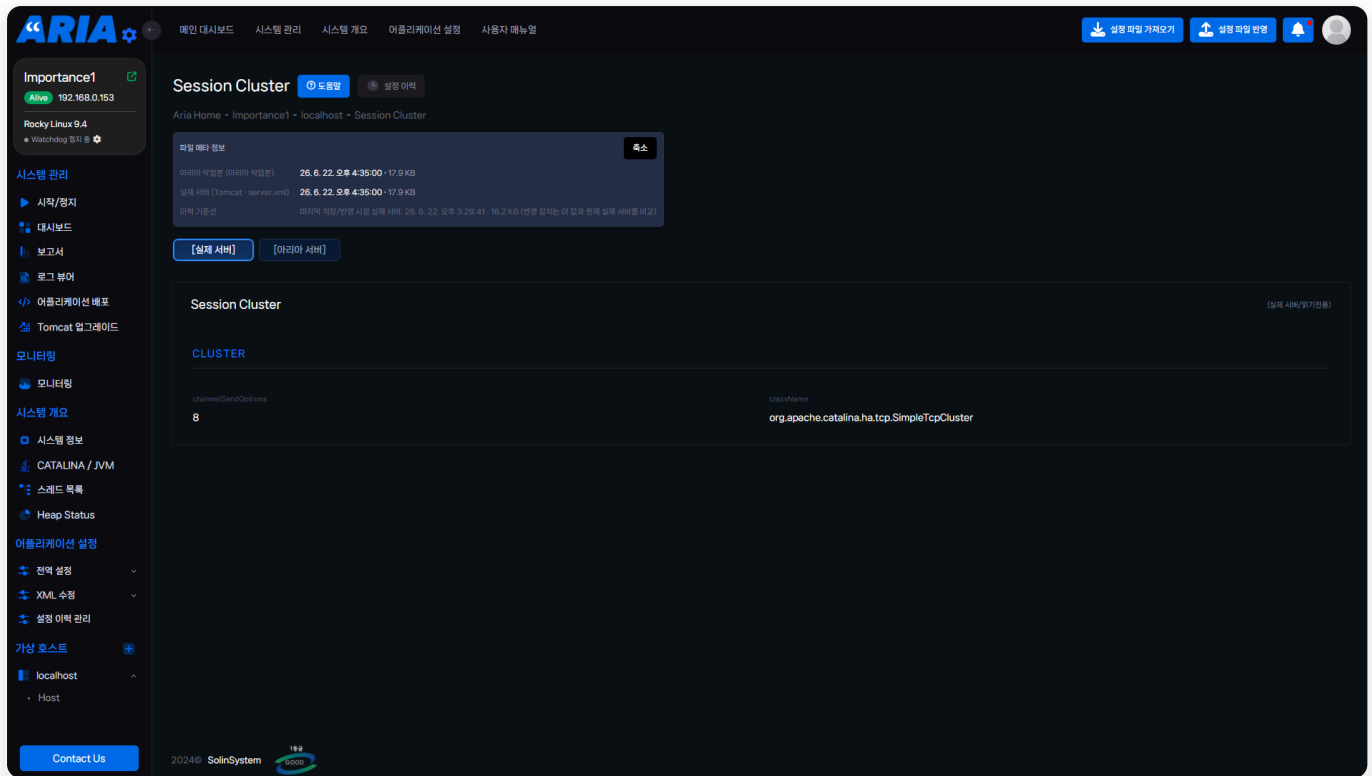
6.25.2 1. 화면 구성

영역	설명
설정 파일 메타	아리아 작업본·실제 server.xml ·이력 기준선의 수정 시각·크기 비교
설정 소스 전환	[실제 서버] / [아리아 서버] 탭
Session Cluster 설정	연결 방식, CLUSTER·MANAGER·RECEIVER·INTERCEPTOR·VALVE·MEMBERSHIP
수정 / 저장 / 취소	편집 제어
설정 이력	server.xml 변경 이력 페이지로 이동

2. 설정 소스 전환

[아리아 서버] 탭은 선택한 vhost 범위의 Session Cluster를 ARIA 작업본에서 편집하는 화면입니다. 기본으로 열리며, **저장** 시 반영 대기 목록에 등록됩니다.

[실제 서버] 탭은 Tomcat에 적용된 vhost 클러스터 설정을 **읽기 전용**으로 조회합니다.



상단 **파일 메타 정보**에서 아리아 작업본과 실제 서버 파일의 최종 수정 시각·크기, 이력 기준선과의 차이를 확인할 수 있습니다.

6.25.3 3. 설정 항목

이 페이지는 특정 **가상 호스트 Host**에 속한 <Cluster> 정의만 편집합니다. Engine 수준 전역 클러스터는 **Cluster 설정**에서 관리하세요.

연결 방식

항목	설명	설정 가이드
Multicast	멀티캐스트 주소를 통해 클러스터 노드 간 메시지를 전송하는 방식입니다. 노드가 동적으로 추가·제거될 때 유용합니다.	MEMBERSHIP 구역(<code>address</code> , <code>port</code> 등)을 설정합니다. Receiver <code>address</code> 에는 <code>auto</code> 또는 유효 IP를 입력합니다.
Static	각 노드의 IP·포트를 명시적으로 지정해 통신하는 방식입니다. 노드 구성이 고정된 환경에 적합합니다.	Static 멤버 목록에 <code>host</code> , <code>port</code> 를 등록합니다. Receiver <code>address</code> 에는 고정 IP를 입력합니다.

Cluster

항목	설명	설정 가이드
<code>channelSendOptions</code>	클러스터 내 데이터 전송 옵션입니다. 전송 신뢰성·복제 횟수 등을 조절해 무결성과 전송 속도를 맞출 수 있습니다.	기본값 8. Tomcat Cluster 문서와 환경에 맞게 조정합니다.

Manager

항목	설명	설정 가이드
<code>expireSessionsOnShutdown</code>	서버가 정상 종료될 때 세션을 만료시킬지 여부입니다. 활성화하면 종료 시 세션 데이터 손실을 방지합니다.	<code>true</code> 권장
<code>notifyListenersOnReplication</code>	세션 복제 시 세션 리스너에게 이벤트를 알릴지 여부입니다. 세션 상태 변경 후처리가 필요할 때 사용합니다.	<code>true</code> 권장

Receiver

항목	설명	설정 가이드
<code>address</code>	클러스터 메시지를 수신할 IP 주소입니다. 지정한 주소를 기준으로 노드 간 통신을 관리합니다.	Multicast: <code>auto</code> 또는 유효 IP. Static: 고정 IP (예: <code>192.168.0.10</code>)
<code>port</code>	클러스터 메시지를 수신할 포트입니다. 이 포트로 노드 간 메시지를 주고받습니다.	기본 4000. 방화벽에서 노드 간 통신을 허용해야 합니다.
<code>selectorTimeout</code>	메시지 수신 선택터의 타임아웃(ms)입니다. 지정 시간 내 메시지를 받지 못하면 타임아웃 처리합니다.	기본 5000. 숫자만 입력

항목	설명	설정 가이드
maxThreads	클러스터 메시지를 병렬 처리할 최대 스레드 수입니다.	기본 6. 트래픽에 따라 조정

Interceptor

항목	설명	설정 가이드
TcpFailureDetector	TCP 연결로 다른 노드의 가용성을 확인하고 네트워크 실패를 감지하는 인터셉터입니다.	클러스터 안정성 확보 시 활성화 권장
MessageDispatchInterceptor	클러스터 노드 간 메시지 전달을 관리하고 전송 중 문제를 처리하는 인터셉터입니다.	일반적으로 활성화
ThroughputInterceptor	클러스터 통신 처리량을 추적·통계 제공하는 인터셉터입니다.	성능 분석이 필요할 때 활성화
TcpPingInterceptor	주기적으로 다른 노드에 TCP 핑을 보내 노드 상태를 확인하는 인터셉터입니다.	장애 조기 감지가 필요할 때 활성화

Valve

항목	설명	설정 가이드
ReplicationValve	요청 처리 시 세션 데이터 변경을 다른 클러스터 노드에 복제하는 밸브입니다.	세션 복제가 필요하면 활성화 (필수에 가까움)
JvmRouteBinderValve	로드밸런서와 협력해 세션을 특정 JVM 노드에 바인딩하는 밸브입니다.	스티키 세션·로드밸런서 연동 시 활성화

Membership (Multicast)

항목	설명	설정 가이드
address	멀티캐스트 주소입니다. 클러스터 노드가 서로를 발견할 때 사용합니다.	기본 228.0.0.4. 네트워크에서 허용된 멀티캐스트 대역 사용
port	멀티캐스트 통신 포트입니다.	기본 45564. 0~65535 범위
frequency	멤버십 정보를 전송하는 주기(ms)입니다.	기본 500

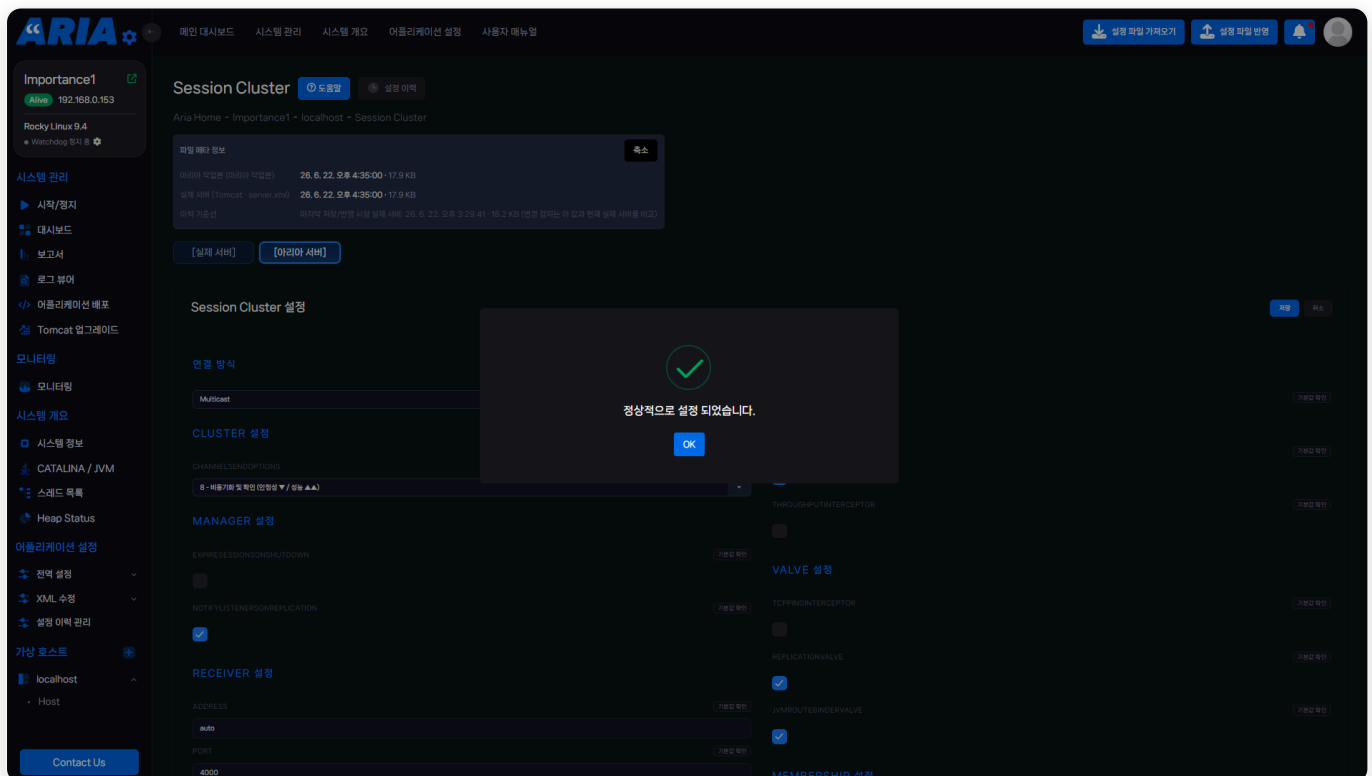
항목	설명	설정 가이드
dropTime	응답이 없는 노드를 클러스터에서 제거하기 전 대기 시간(ms)입니다.	기본 3000

Static 멤버

항목	설명	설정 가이드
host	정적 멤버로 추가할 노드의 호스트명 또는 IP 주소입니다.	유효한 IP 또는 도메인 입력
port	정적 멤버와 통신할 포트입니다.	0~65535 범위
uniqueId	정적 멤버를 클러스터 내에서 고유하게 식별하는 ID입니다.	노드 간 충돌 방지를 위해 고유값 지정

6.25.4 4. Cluster 설정 편집

1. [아리아 서버] 탭에서 **수정** 버튼을 클릭합니다.
2. **설정 항목**을 참고해 **연결 방식**(Static / Multicast)과 각 구역 속성을 변경합니다.
3. **저장**을 클릭하면 아리아 작업본에 반영되고 반영 대기 목록에 등록됩니다.



버튼	동작
수정	Session Cluster 편집 모드 진입
저장	아리아 작업본에 저장 (반영 대기)
취소	편집 중 변경 사항 되돌림

주의: **저장**은 즉시 Tomcat에 적용되지 않습니다. 헤더 **설정 파일 반영**으로 실제 서버에 적용해야 합니다.

6.25.5 5. 서버 반영

1. 설정을 저장하면 헤더 우측 **반영 대기** 배지에 변경 건수가 표시됩니다.
2. **설정 파일 반영**으로 대상 항목을 선택해 반영합니다.

자세한 절차는 [설정 변경 반영](#) 문서를 참고하세요.

6.25.6 💡 활용 팁

- **전역 vs vhost:** 서버 전체 클러스터는 [Cluster 설정](#)에서 관리하세요.
- **실제 서버와 비교:** 반영 전 [실제 서버]·[아리아 서버] 탭을 번갈아 확인하세요.
- **Receiver 주소:** Multicast 모드에서는 address 에 auto 또는 유효한 IP를 입력해야 합니다.
- **설정 이력:** 페이지 제목 옆 **설정 이력** 버튼으로 변경 이력을 조회·복원할 수 있습니다. ([설정 변경 이력](#))