The White Book of GNX AAL

검증·라이선스 계약용 공식 백서

목적 결속형 실행 객체 비생성 제어 기반 접근·세션·실행·반출 및 도구 호출 통제 체계

공식 엔터프라이즈 문서 | 2026. 6. 12. 기준 정합본

문서 등급	Enterprise / Official Reference
문서 성격	검증·라이선스 계약용 공식 백서
작성 목적	리더, 보안책임자, 플랫폼 책임자, 법무·라이선싱 담당자, 기술실사 담당자가 GNX AAL의 기술적 본질과 검증 기준을 동일한 언어로 평가하도록 하는 정합본
정합 기준	명세서, 도 1~8, 선행기술진보성조사보고서, 선행기술대비표, 우선심사신청설명서, AAL 기술자료, 운영 PoC 측정자료, 사업행정자료 및 보조자료
핵심 명제	non_birth = issued_objects 실행 가능 네임스페이스 미생성
표현 원칙	전 세계 전수 부존재를 단정하지 않고, 조사 범위와 기준일 하에서 확인된 자료의 단독 또는 결합으로부터 핵심 구성이 개시·시사되지 않는다는 한정형 표현을 사용함

본 정합본의 중심 문장

본 백서의 핵심은 GNX AAL을 기능 목록이 아니라 발급 제어 평면과 non_birth 저장구조 불변식으로 설명하는 데 있다. non_birth는 거절 메시지가 아니라, issued_objects 실행 가능 네임스페이스에 후단 실행 가능한 객체 참조값이 생성되지 않는 사실 상태이다.

GNX AAL

목차

본 목차는 백서의 존재 이유에 맞추어 라이선싱·실사·선행기술 대비·경영 판단·검증 프로토콜 순서로 구성하였다.

공식 의견서 서문

본 백서는 기존 청서·백서 업로드본을 전제로 한 단순 교정본이 아니라, 명세서, 도면, 선행기술 진보성 자진조사보고서, 선행기술대비표, 우선심사신청설명서, 기술자료, 운영 PoC 측정자료, 사업행정자료 및 보조자료의 핵심 논리와 용어를 통합한 2026년 6월 12일 기준 정합본이다.

GNX AAL의 중심 문장은 하나이다. 고위험 행위 요청에 대하여 목적 객체, 범위 해시, 활성 정책 버전, 감사 사전기록 및 발급 트랜잭션이 성공적으로 결합되지 않으면 후단 실행 가능한 객체 참조값 자체가 실행 가능 네임스페이스에 생성되지 않는다. 따라서 본 발명은 더 강한 거절, 더 자세한 로그, 더 늦은 폐기가 아니라 실행 객체의 출생 자체를 통제하는 구조로 이해되어야 한다.

본 백서는 절대적·전칭적 부존재를 단정하지 않는다. 조사 범위와 기준일 하에서 확인된 선행문헌의 단독 또는 결합으로부터 목적 객체·범위 해시·활성 정책 버전·감사 사전기록·상태기계·네임스페이스 분리·verify/consume 후단 강제의 결합이 개시 또는 시사되지 않는다는 한정형 표현을 사용한다. 이 표현은 약한 표현이 아니라 심사와 실사에서 신뢰성을 높이는 공식 문체이다.

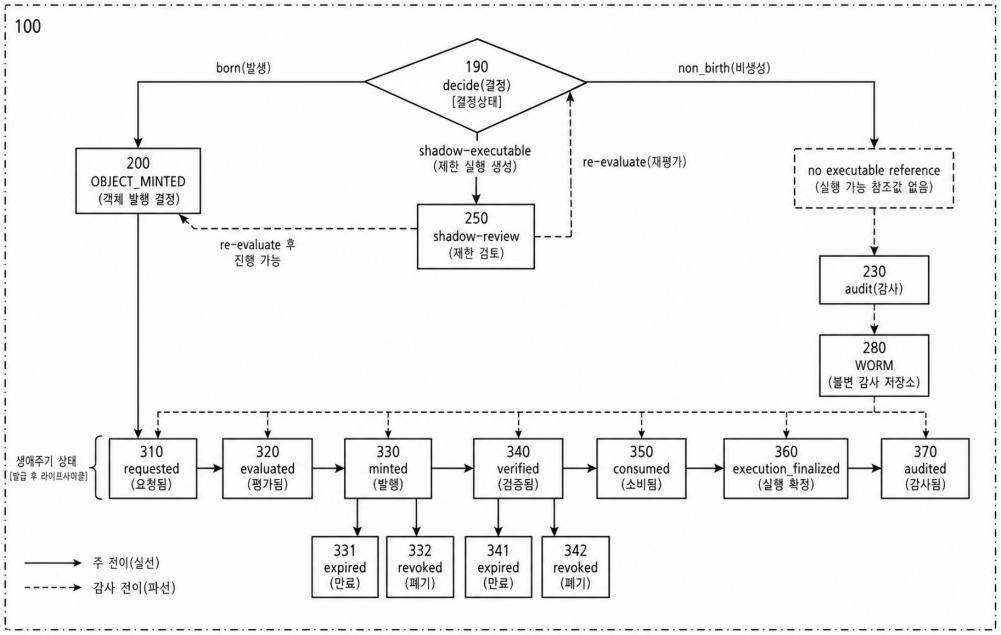
문서 성격 및 표현 원칙

본 문서는 기술·라이선스·실사 정리본이며, 특허성·침해성·무효가능성에 관한 최종 법률 판단은 등록된 변리사 또는 대리인의 검토를 전제로 한다.

요약문

백서는 GNX AAL을 경영진·법무·보안전문가가 검증 가능한 구조로 이해하도록 만드는 문서이다. 핵심 평가지점은 정책 판단의 존재가 아니라, 정책 판단이 실행 가능 객체의 존재 또는 부존재와 어떤 트랜잭션 경계에서 결합되는가이다.

제1장 GNX AAL 백서의 목적과 핵심 명제



도 3

그림 1. 객체 발생 결정과 생애주기 상태: requested, evaluated, minted, verified, consumed, finalized, audited

1.1 라이선싱 관점의 핵심 정의

GNX AAL의 라이선싱 대상은 단일 API, 단일 토큰 발급기 또는 프론트엔드 정책 화면이 아니다. 라이선싱 대상은 고위험 행위 요청을 목적 결속형 실행 객체 발생 요청으로 정규화하고, 발급 전제절차가 성공하지 않으면 실행 가능 네임스페이스에 후단 실행 가능한 객체 참조값을 생성하지 않으며, 생성된 객체도 verify와 consume을 통과하기 전에는 후단 실행되지 않도록 하는 발급 제어 평면이다.

발급 제어 평면은 요청 수신, 주체 확인, 기준 구조체 참조, 증거 형성, 목적 결속, 범위 해시, 정책 버전 결합, 감사 사전기록, 객체 발생 결정, 원자적 발급, 검증, 소비 트랜잭션, 감사 체인, 후단 강제를 하나의 상호의존적 체계로 묶는다.

1.2 non_birth의 실무적 의미

non_birth는 deny, revoke, blacklist, inactive flag, UI 거절 또는 사후 감사로그와 다르다. deny는 요청을 거절한다는 의미이고, revoke는 이미 존재하는 권능을 폐기한다는 의미이며, inactive는 검증 시점의 상태값이다. 반면 non_birth는 실행 가능한 issued_objects 행, object_jti, signed capability, verification_ref, session handle, export handle, file creation handle, BI extract reference, tool invocation id 또는 execution handle이 실행 가능 저장구조에 생성되지 않는 결정상태이다.

이 차이는 공격 표면에 직접 연결된다. 이미 만들어진 객체가 없다면 복제, 캐시 노출, 로그 노출, 직접 DB 접근, 배치 경로, BI 커넥터 경로 또는 이전 메시지 재사용을 통한 후단 재실행 가능성이 구조적으로 축소된다.

1.3 리더가 이해해야 할 가치 명제

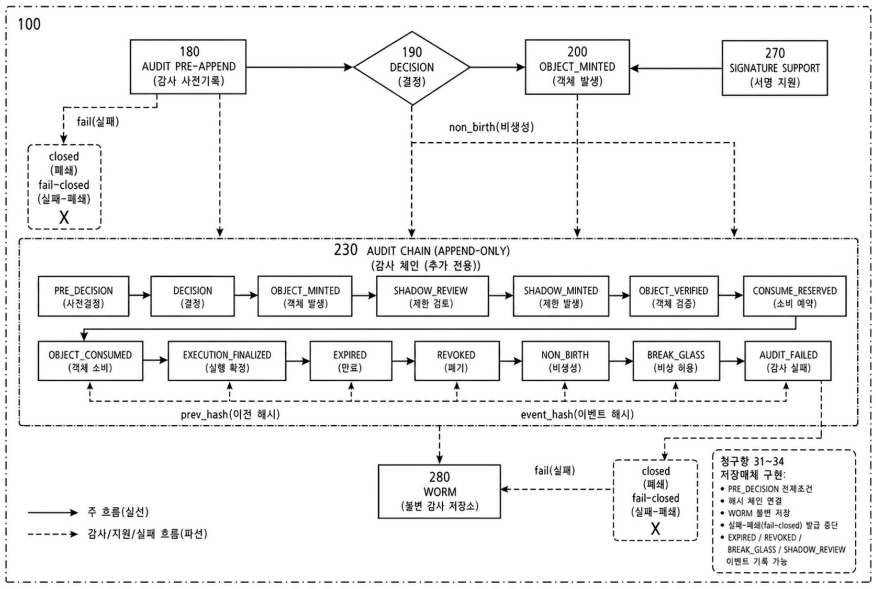
관점	기존 보안 언어	GNX AAL 언어
정책 판단	허용 또는 거절	실행 객체가 태어날 수 있는지 여부
감사	사후 기록 및 추적	발급 전제조건 및 해시 체인
토큰	발급 후 active 검증	조건 미충족 시 토큰 자체 미생성
반출	파일 생성 후 탐지·차단	파일·URL·반출 핸들 미생성
AI 도구 호출	프롬프트 또는 tool list 제한	tool invocation id 자체의 출생 통제

1.4 백서의 검증 기준

- non_birth 결정 시 issued_objects 실행 가능 네임스페이스에 대응 행이 없어야 한다.
- audit_trace_id, request_fingerprint, non_birth_event_id는 object_id, object_jti 또는 verification_ref로 해석되지 않아야 한다.
- verify(non_birth audit id)는 active를 반환하지 않아야 하며 unknown, inactive 또는 non_executable 류의 실패 상태만 반환해야 한다.
- 후단 실행 시스템은 verify 및 consume을 통과하지 않은 식별자만으로 실행하지 않아야 한다.
- 감사 사전기록, 감사 완료기록, 감사 해시 체인 갱신 또는 WORM 기록이 실패하면 born 및 shadow-executable 발급은 fail-closed로 중단되어야 한다.

> **요약문**
>
> 제1장은 GNX AAL의 라이선싱 대상을 발급 제어 평면으로 특정하였다. 검증 기준은 거절 표시가 아니라 실행 가능 객체의 미생성, 검증 active 부재, 후단 실행 차단, 감사 실패 시 실패-폐쇄를 객관적으로 입증하는 것이다.

제2장 선행기술 대비와 구조적 차별성



도 7

그림 2. 감사 체인 및 WORM 구조: PRE_DECISION, NON_BIRTH, AUDIT_FAILED와 fail-closed 흐름

2.1 종래 접근제어와의 차이

PBAC와 ABAC는 목적, 속성, 정책 조건을 접근 판단에 반영한다. Zero Trust는 지속 검증과 최소 권한을 강조한다. OAuth 계열 토큰 검증은 이미 발급된 토큰의 active 상태 또는 보유자 조건을 검증한다. PAM은 특권 세션을 기록하고, DLP는 생성되었거나 이동하려는 데이터 내용을 탐지한다. 멱등키와 행 잠금은 중복 처리와 경쟁 조건을 줄인다. 이 기술들은 중요하지만, 목적 객체·범위 해시·정책 버전·감사 사전기록을 실행 가능 객체의 행 생성 여부와 하나의 발급 트랜잭션 경계에서 결합하는 구조까지 보장하지는 않는다.

2.2 존재하지만 전달되지 않음과 존재 자체가 없음

라이선싱 검증에서 가장 큰 오해는 trustee 기반 secret release, 토큰 발급 후 폐기, 부분허용 객체, 샌드박스 실행을 non_birth와 동일시하는 것이다. 그러나 존재하지만 전달되지 않는 비밀, 발급되었지만 비활성화된 토큰, 프론트엔드가 가린 다운로드 URL, tool 목록에서 숨겨진 tool은 여전히 탐색·복제·로그·캐시·우회 가능성을 남긴다. non_birth는 실행 참조값을 생성하지 않는다.

2.3 선행기술 대비의 객관적 축

구성 축	GNX AAL의 정합 구성	선행기술 대비 차이
목적 객체	object_type, action, resource_ref, normalized_scope, risk_class, approval_ref, retention_rule, evidence_validity를 포함하는 기계 검증 구조체	목적·의도·작업 신호는 있어도 행 생성/미생성의 원자 결합은 다름
범위 해시	후단 재검증 가능한 normalized_scope hash	scope caveat 또는 최소 권한과 달리 미생성 선행조건으로 결합
감사 사전기록	PRE_DECISION 성공을 발급 전제조건으로 사용	사후 감사 또는 append-only 로그와 시점이 다름
상태기계	born, shadow-executable, shadow-review, non_birth	단순 allow/deny 또는 sandbox block과 행 존재 여부가 다름
후단 강제	verify, consume, backend enforcement, WORM	토큰 active 확인 또는 실행 후 증명만으로는 부족

2.4 결합발명 방어의 중심 논리

복수 선행기술을 결합하더라도 남는 핵심은 네 가지이다. 첫째, 목적 객체·범위 해시·활성 정책 버전·감사 사전기록의 발급 전제조건 결합이다. 둘째, born, shadow-executable, shadow-review, non_birth로 분기되는 상태기계이다. 셋째, issued_objects와 non_birth_events의 네임스페이스 분리 및 실행 가능 행 미생성이다. 넷째, verify, consume, backend enforcement의 후단 강제 결합이다. 이 네 요소가 동시에 충족되어야 non_birth가 단순 거절이 아니라 저장구조상 비생성이라는 기술적 상태가 된다.

> **요약문**
>
> 제2장은 기존 보안기술을 부정하지 않고 그 한계를 구분하였다. GNX AAL의 차별점은 각 요소의 병렬 나열이 아니라, 발급 성패와 객체 존재 여부를 결합하는 제어 지점에 있다.

제3장 라이선스·검증 기준

3.1 라이선스 계약에서의 대상 특정

계약서 또는 검증 부속서에서 GNX AAL을 특정할 때에는 “접근제어 솔루션”이라는 넓은 표현만으로는 부족하다. 적절한 대상 특정은 “목적 결속형 실행 객체 발생 요청을 정규화하고, 발급 전제절차가 성공하지 않으면 실행 가능 네임스페이스에 후단 실행 가능한 객체 참조값을 생성하지 않으며, 생성된 객체도 검증 및 소비 트랜잭션을 통과하기 전에는 실행되지 않도록 하는 발급 제어 평면”이다.

3.2 필수 검증 산출물

산출물	검증 목적
구현 설명서	요청 정규화부터 후단 강제까지 제어 경계 확인
API 계약서	/decide, /verify, /consume, /audit의 입력·출력 및 실패 상태 확인
저장구조 스키마	issued_objects, non_birth_events, audit_events 등 네임스페이스 분리 확인
상태전이 표	born, shadow-executable, shadow-review, non_birth의 의미와 전환 조건 확인
감사 체인 명세	prev_hash, event_hash, WORM, fail-closed 조건 확인
검증 시나리오 결과	non_birth 7 조건, replay 차단, tenant isolation, audit chain integrity 확인
후단 강제 연동 증거	verify/consume 실패 시 실제 backend execution 차단 확인

3.3 수용 시험: non_birth 7 조건

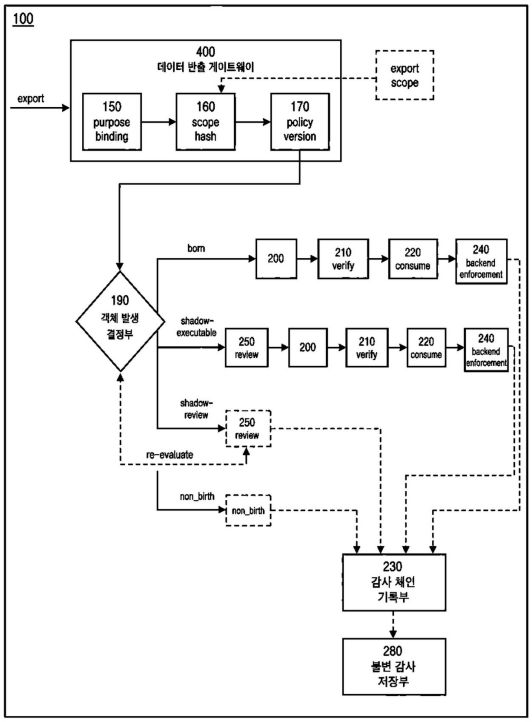
시험 조건	기대 결과
목적 없음 또는 목적 객체 불일치	object_token, handle, verification_ref 미생성
empty scope 또는 범위 해시 불일치	verify active 부재 및 backend 차단
evidence 만료	non_birth 이벤트만 생성, 실행 참조값 미생성
정책 버전 비활성 또는 배포 epoch 불일치	born/shadow-executable 발급 중단
감사 사전기록 실패	fail-closed 및 신규 실행 가능 행 미생성
서명키 준비 실패 또는 capability 서명 실패	외부 반환 금지 및 non_executable 처리
테넌트 불일치 또는 RLS 실패	tenant boundary 위반으로 active 반환 금지

3.4 라이선싱 리스크와 제한 문구

본 백서는 기술적 주장 문서이지 최종 법률의견이 아니다. 라이선스 계약에는 제공 문서 및 테스트 결과에 기초하여 구현 수준을 평가한다는 점, 미공개 출원 또는 색인 누락 문헌의 존재 가능성은 별도 실사 대상이라는 점, 특허성·침해성·무효가능성에 대한 최종 판단은 대리인 검토를 전제로 한다는 점을 명확히 기재하는 것이 적절하다.

> **요약문**
>
> 제3장은 계약과 실사에서 사용할 수 있는 검증 산출물과 수용 시험을 제시하였다. 핵심은 non_birth 7 조건, 네임스페이스 분리, verify active 부재, 후단 실행 차단, 감사 실패 시 fail-closed 발급 중단이다.

제4장 사례 기반 fact 분석



도 6

그림 3. 데이터 반출 게이트웨이: purpose binding, scope hash, policy version과 non_birth 감사 흐름

4.1 금융권 고객정보 대량조회

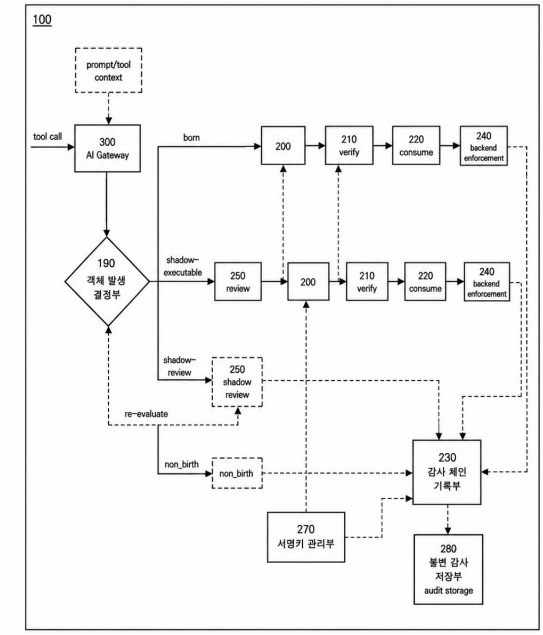
이미 인증된 직원이 고객정보 대량조회를 요청하는 상황에서 종래 접근제어는 역할, 부서, 승인 여부, 시간대, 단말, 이상행위 점수 등을 평가한다. GNX AAL은 여기에 더해 대량조회 요청을 실행 객체 발생 요청으로 정규화한다. 목적 객체는 사건번호, 업무티켓, 승인 참조, 보존 규칙, 증거 유효성과 결속된다. 범위 해시는 고객 범위, 필드 집합, 행 수, 마스킹 프로파일, 스키마 버전으로 산출된다. 조건이 실패하면 조회 실행 핸들이 생기지 않는다.

4.2 데이터 반출 게이트웨이

반출 요청은 파일 생성 핸들, 다운로드 URL, 외부 전송 핸들, BI 추출 참조값이 발생하는 순간 위험해진다. GNX AAL은 반출 목적, export scope, 범위 해시, 정책 버전을 결합하고 결정부가 born, shadow-executable, shadow-review, non_birth로 분기한다. non_birth에서는 파일을 먼저 만들고 URL만 숨기는 것이 아니라 파일 생성 참조값 또는 외부 전송 참조값이 실행 가능 네임스페이스에 생성되지 않아야 한다.

4.3 AI 에이전트 도구 호출

AI 에이전트가 외부 API, 결제, 파일 쓰기, 데이터베이스 쿼리, 배포 도구를 호출하려는 상황에서는 prompt 안전성만으로 충분하지 않다. GNX AAL은 tool call을 독립 실행 객체 발생 요청으로 변환한다. AI Gateway는 prompt/tool context를 받아 목적 결속형 실행 객체 발생 요청으로 정규화하고, 결정부가 비생성을 산출하면 tool invocation id 또는 execution handle이 생성되지 않는다.



도 5

그림 4. AI Gateway에서 prompt/tool context를 실행 객체 발생 요청으로 정규화하는 흐름

4.4 관리자 예외 및 break-glass

엔터프라이즈 환경에는 긴급 장애 대응, 법적 보전, 운영자 예외 처리가 존재한다. GNX AAL은 이를 배제하지 않고 break-glass와 shadow 상태로 흡수한다. 중요한 점은 예외 자체도 감사 체인과 정책 버전에 결합되어야 하며, 예외 승인 후에도 검증과 소비 트랜잭션을 우회해서는 안 된다는 점이다. 예외는 객체 발생 결정의 입력 조건일 수 있으나 후단 실행 가능 객체가 임의로 태어나는 면허가 아니다.

> **요약문**
>
> 제4장은 금융권 대량조회, 데이터 반출, AI 에이전트 도구 호출, 관리자 예외를 통해 GNX AAL의 통제 지점을 설명하였다. 공통 fact는 실행 참조값이 태어나는 시점이 위험의 전환점이며, GNX AAL은 그 전환점을 발급 트랜잭션 경계에서 통제한다는 것이다.

제5장 엔터프라이즈 실사 프레임

5.1 검증팀을 위한 실사 질문

질문	합격 기준
정책 평가와 발급 객체 행 생성이 같은 트랜잭션 경계에서 결합되는가	목적·범위·정책·감사 실패 시 issued_objects 행 미생성
non_birth 이벤트가 후단 실행 입력으로 해석될 수 있는가	감사 식별자와 실행 식별자가 명확히 분리됨
shadow-review가 실행 핸들을 포함하는가	검토 레코드만 존재하고 실행 참조값 없음
shadow-executable 제한이 후단에서 강제되는가	restriction_flags, expires_at, scope_hash, usage_policy가 verify/consume에 반영
감사 실패 시 시스템이 fail-open 하는가	신규 born 및 shadow-executable 발급 중단

5.2 증거 패키지 구성

실사 패키지는 기술 설명, API 계약, 저장구조 스키마, 테스트 결과, 운영 PoC, 감사 체인 증거, 사업 행정 자료를 분리하여 제공하는 편이 적절하다. 기술자료는 명세서/도면 대응을 설명하고, 운영 PoC는 aal.kr 공개 사이트, OS Engine, scenario routing, decision/verify/consume/replay, invariant proof 및 audit chain을 증거화한다. 사업행정자료는 실시준비 주체와 온라인 제공 기반을 설명한다.

5.3 성숙도 단계

단계	상태	판단 기준
L1	정책 판단 단계	allow/deny가 있으나 실행 객체 출생 제어 없음
L2	발급 결합 단계	목적·범위·정책이 발급 조건으로 일부 결합
L3	non_birth 단계	조건 실패 시 실행 가능 네임스페이스 미생성
L4	후단 강제 단계	verify/consume/backend enforcement까지 일관
L5	감사 불변 단계	PRE_DECISION부터 WORM까지 fail-closed로 연결

5.4 계약 부속 검증 문구 예시

> **부속 검증 문구**
>
> 수요자는 non_birth 결정상태에서 object_id, object_jti, signed capability, verification_ref, session handle, export handle, file creation handle, tool invocation id 및 execution handle이 issued_objects 또는 이에 준하는 실행 가능 네임스페이스에 생성되지 않음을 API 응답, 저장구조, 로그, 메시지 버스, 캐시 및 후단 실행 연동 증거로 확인한다.

> **요약문**
>
> 제5장은 실사 질문, 증거 패키지, 성숙도 단계, 계약 부속 문구를 제시하였다. 실사는 기능 체크리스트가 아니라 실행 참조값의 존재 여부를 확인하는 저장구조·트랜잭션 검증이어야 한다.

제6장 최종 의견 및 라이선싱 결론

6.1 경영진 결론

경영진 관점에서 GNX AAL은 보안 도구 하나를 추가하는 것이 아니라, 고위험 행위가 현실화되는 순간을 통제하는 제어 평면이다. 고객정보 대량조회, 반출, 운영 콘솔, AI 도구 호출, 배치 실행은 모두 실행 참조값이 발생하는 순간 위험해진다. GNX AAL은 이 발생 시점을 경영진의 리스크 정책과 감사 체인에 연결한다.

6.2 보안전문가 결론

보안전문가 관점에서 GNX AAL은 기존 PBAC, ABAC, OAuth, PAM, DLP, 멱등 트랜잭션, WORM 감사를 폐기하는 기술이 아니다. 각 기술을 발급 제어 평면 안으로 수렴시켜, 목적·범위·정책·감사·검증·소비·후단 강제를 동일한 불변식 아래 운용하게 하는 구조이다.

6.3 라이선싱 결론

라이선싱 관점에서 핵심 가치는 “non_birth를 구현했는가”이다. 표현상 non_birth라는 단어를 쓰더라도 실행 가능 객체가 내부에서 생성된 뒤 전달만 막히는 구조라면 GNX AAL의 핵심과 다르다. 반대로 용어가 다르더라도 목적 결속형 실행 객체 발생 요청, 발급 전제조건, 네임스페이스 미생성, verify/consume 후단 강제, 감사 fail-closed가 모두 구현되어 있다면 실질적으로 GNX AAL의 평가 대상이 된다.

6.4 최종 의견

GNX AAL의 기술적 명확성은 non_birth라는 하나의 단어가 아니라, 그 단어를 저장구조, 상태기계, API, 트랜잭션, 후단 실행, 감사 체인에서 동시에 관찰할 수 있다는 점에서 발생한다. 따라서 본 백서의 최종 판단 문장은 다음과 같다. GNX AAL은 실행 가능 객체의 출생을 목적·범위·정책·감사 전제조건과 원자적으로 결합하고, 조건 미충족 시 실행 가능 네임스페이스에 객체 참조값이 형성되지 않도록 하는 발급 제어 평면이다.

> **요약문**
>
> 제6장은 경영진, 보안전문가, 라이선싱 담당자의 결론을 통합하였다. GNX AAL의 가치는 기존 보안 요소의 단순 합이 아니라 실행 객체 출생 통제를 중심으로 한 구조적 결합에 있다.

제7장 라이선스 부속 검증 프로토콜

7.1 검증 프로토콜의 원칙

검증 프로토콜은 화면 캡처 중심이 아니라 저장구조·API·감사 체인·후단 실행을 함께 확인하는 방식이어야 한다. 특히 non_birth가 표시되는 UI만으로는 충분하지 않다. 동일 요청에 대해 응답, issued_objects, non_birth_events, audit_events, cache, message bus, backend enforcement 결과를 교차 확인해야 한다.

7.2 증거의 보존 방식

검증 과정에서 생성된 응답 JSON, DB 스냅샷, 감사 이벤트 해시, WORM 저장 증거, 후단 실행 차단 로그는 같은 case_id 또는 audit_trace_id로 묶어 보존한다. 단, audit_trace_id는 실행 참조값이 아님을 명시하고 verify 입력으로 사용될 때 active가 반환되지 않는지 확인한다.

7.3 계약상 인수 조건

인수 조건	측정 방법
non_birth 7 조건 통과	각 실패 조건별 object/handle 미생성 및 verify active 부재 확인
born 정상 경로 통과	issued_objects 행 생성, verify active, consume finalize 확인
shadow-executable 제한 강제	제한 플래그·만료·범위 축소·사용횟수 제한 확인
shadow-review 비실행성	검토 레코드만 존재하고 실행 참조값 미생성 확인
감사 실패 fail-closed	감사 기록 실패 시 신규 발급 중단 확인

7.4 침해·회피 검토의 리트머스

회피 설계 검토의 핵심 질문은 “객체를 만들지 않았는가”이다. 이름을 바꾸거나 토큰이 아닌 handle, URL, job id, tool call id, verification reference를 쓰더라도 후단 실행 가능한 참조값이 조건 실패 시 생성된다면 non_birth가 아니다. 반대로 어떤 명칭을 쓰더라도 조건 실패 시 실행 가능 네임스페이스에 대응 참조값이 없고, 감사 식별자도 실행 입력이 되지 않으며, 후단 verify/consume이 이를 강제한다면 GNX AAL의 핵심 구조에 가깝다.

> **요약문**
>
> 제7장은 라이선스 부속 검증 프로토콜을 제시하였다. 검증은 UI 표시가 아니라 데이터 모델, API 응답, 후단 실행 차단, 감사 체인을 통합하여 관찰해야 한다.

제8장 심사·실사 대응 결론

8.1 심사·실사 대응 관점

심사·실사 대응에서 GNX AAL은 “정책 기반 접근제어”라는 넓은 범주로만 설명되어서는 안 된다. 반드시 실행 객체 발생 요청, 목적 객체, 범위 해시, 활성 정책 버전, 감사 사전기록, 객체 발생 결정, 원자적 발급, 검증, 소비, 후단 강제, 감사 체인이라는 순서로 설명해야 한다. 이 순서가 무너지면 non_birth는 단순 거절 플래그로 오해될 수 있다.

8.2 객관적 근거 문장

> **공식 설명 문장**
>
> GNX AAL은 고위험 행위 요청을 목적 결속형 실행 객체 발생 요청으로 정규화하고, 목적 객체·범위 해시·활성 정책 버전·감사 사전기록을 발급 전제조건으로 결합하며, 조건 미충족 또는 발급 트랜잭션 실패 시 issued_objects 실행 가능 네임스페이스에 후단 실행 가능한 객체 참조값 자체가 생성되지 않는 non_birth 저장구조 상태를 형성한다.

8.3 기업 도입 효과

도입 영역	기대 효과
고객정보 조회	목적·범위·증거 불일치 시 조회 핸들 미생성
데이터 반출	파일·URL·export handle의 출생 전 통제
AI 도구 호출	tool invocation id 및 execution handle 미생성
운영자 예외	break-glass도 감사 체인과 정책 버전에 결합
배치/서비스 계정	사람 계정 외 자동화 경로에도 동일한 발급 제어 적용

8.4 최종 정리

GNX AAL은 이미 존재하는 객체의 사후 검증·폐기·감사를 강화하는 기술이 아니라, 객체의 출생 여부를 발급 저장구조와 트랜잭션 경계에서 통제하는 기술이다. 심사와 실사에서 이 문장을 일관되게 유지하는 것이 가장 중요하다.

> **요약문**
>
> 제8장은 심사·실사 대응 문장과 기업 도입 효과를 정리하였다. GNX AAL 설명의 중심은 non_birth = 실행 가능 네임스페이스 미생성이라는 구조적 사실이다.

제9장 임원·보안 아키텍처 보고 기준

9.1 임원 보고용 판단 기준

임원 보고에서는 GNX AAL을 “권한을 더 세밀하게 나누는 도구”가 아니라 “고위험 실행 객체가 태어나는 순간을 통제하는 제어 평면”으로 설명한다. 투자 판단 기준은 정책 룰 수가 아니라 non_birth 불변식의 구현 정도, 후단 강제 연동 수준, 감사 실패 시 fail-closed 능력, 데이터 반출 및 AI 도구 호출 적용 범위이다.

9.2 보안 아키텍처 검토 기준

보안 아키텍트는 AAL을 인증 시스템 뒤, 정책엔진 앞, 후단 실행 전 단계에 위치시키는 것이 아니라, 발급 저장구조와 후단 실행 계약을 함께 설계해야 한다. issued_objects는 실행 가능 네임스페이스이고, non_birth_events는 비실행 감사 네임스페이스이며, audit_events는 추가기록 전용 감사 체인이다. 이 세 계층이 혼합되면 핵심 불변식이 손상된다.

9.3 공식 설명의 문체 원칙

- “전 세계에 없다”는 식의 전칭 표현을 피하고, 조사 범위와 기준일 하에서 확인되지 않았다는 한정형 표현을 사용한다.
- “차단한다”보다 “실행 가능 객체 참조값이 생성되지 않는다”는 구조적 표현을 우선한다.
- “토큰 검증 강화”가 아니라 “검증 대상 토큰 자체의 미생성”이라고 구분한다.
- “감사로그를 남긴다”가 아니라 “감사 사전기록이 발급 전제조건이다”라고 설명한다.
- “AI guardrail”이 아니라 “tool call을 실행 객체 발생 요청으로 정규화하고 tool invocation id의 출생을 통제한다”라고 설명한다.

9.4 최종 보고 문장

최종 보고 문장

GNX AAL은 고위험 행위의 실행 가능 객체가 태어나는 순간을 목적, 범위, 증거, 정책 버전, 감사 사전기록, 발급 저장구조, 검증, 소비 트랜잭션 및 감사 체인에 결합하여 통제하는 발급 제어 평면이다. 따라서 조건 미충족 시 남는 것은 실행 가능한 객체가 아니라 비실행 감사 사실이며, 후단 시스템은 그 감사 사실을 실행 참조값으로 사용할 수 없다.

요약문

제9장은 임원 보고와 보안 아키텍처 검토 기준을 제시하였다. 최종 설명은 “더 좋은 거절”이 아니라 “실행 객체 출생 통제”라는 문장으로 수렴해야 한다.