기업 내 AI Proxy 아키텍처 및 보안 통제 정책

# 1. AI Proxy 아키텍처 구성

AI Proxy 아키텍처는 다음 두 가지 목적을 중심으로 설계됩니다:  
① AI 사용을 통제하고, ② 보안 위험을 줄이며, 로그를 남겨 추적 가능하게 한다.

## 1.1 아키텍처 구성도

[사용자 PC] → [AI Proxy 서버] → [외부 AI API]  
- 사내 보안 Agent: 사용자 PC에 설치된 보안 감시 도구  
- 입력/출력 필터링: 민감 정보 필터링 및 마스킹  
- 사용로그 기록: 감사 로그 저장 및 정책 엔진 연동

## 1.2 주요 기능

1) 인증 및 권한 확인: 사용자 인증 후 사용 허용  
2) 입력 필터링: 민감정보 자동 탐지 및 경고/차단  
3) 출력 필터링: AI 응답의 민감성 검토  
4) 마스킹 처리: 이름/이메일 등 자동 마스킹  
5) 로그 기록: 입력, 출력, 시간, 사용자 정보 저장  
6) 정책 엔진: 부서/직무별 허용 범위 설정  
7) 허용 모델 관리: 사용 가능한 AI 목록 지정

## 1.3 기술 스택

- Web Proxy: Nginx + Lua / Envoy  
- 인증 방식: OAuth2, SAML, LDAP 연동  
- 필터링: RegEx, NLP 기반 민감정보 탐지  
- 로그: ELK Stack (Elasticsearch, Logstash, Kibana)  
- 배포: On-Prem 또는 사내 클라우드 (Docker/Kubernetes)

## 1.4 PoC 시나리오

1. 사용자가 ChatGPT 사이트 접속  
2. 프록시 서버가 요청 가로채기  
3. 주민번호 탐지 시 "[MASKED]" 처리  
4. 입력/응답 모두 로그로 저장  
5. 관리자 대시보드에서 사용 모니터링

# 2. AI 사용 정책 및 가이드라인

## 2.1 정책 목적

- 기업 내 AI 도구 사용을 통제하고, 정보 유출을 방지하며, 업무 효율성을 유지함  
- Shadow AI를 탐지하고, 내부 AI 거버넌스 프레임워크 확립

## 2.2 정책 내용

- AI 사용 시 승인 프로세스 운영 (사전 신청제)  
- 부서별 허용 AI 목록 지정 (예: R&D팀만 Copilot 허용)  
- 민감정보 입력 금지 정책 고지 및 교육  
- 주기적 사용 내역 점검 및 이상 탐지 보고

## 2.3 역할 및 책임

- 사용자: AI 사용 시 정책 준수  
- IT보안팀: Proxy 관리, 로그 분석, 리포트 제출  
- 부서장: AI 사용 승인 및 사용자 교육 관리  
- 감사팀: AI 사용 감사 및 규정 준수 평가

# 3. Shadow AI 탐지 체계

## 3.1 정의 및 위험성

Shadow AI란 기업 내 공식적인 승인 없이 개별 사용자가 무단으로 사용하는 생성형 AI 도구 및 서비스를 의미합니다.  
이는 다음과 같은 보안 위협을 초래할 수 있습니다:  
- 민감 정보 유출 위험 증가  
- 비인가 외부 API 사용으로 인한 데이터 통제권 상실  
- AI 생성 결과에 대한 오류 책임 불분명  
- 컴플라이언스 미준수 (예: ISMS-P, GDPR 위반)

## 3.2 탐지 전략

1) 네트워크 트래픽 분석  
- AI 관련 도메인 접근 탐지 (예: \*.openai.com, \*.anthropic.com)  
- HTTPS SNI 및 DNS 로그 분석 기반 접근 기록 확보  
  
2) 엔드포인트 모니터링  
- PC/노트북 내 AI 앱 설치 여부 감지 (예: Copilot Desktop, AI Notepad 등)  
- 브라우저 확장 프로그램 모니터링 (GPT 플러그인 등)  
  
3) 로그 기반 이상 행동 탐지  
- 업무 시간 외 AI 사용 기록  
- 과도한 텍스트 복사/붙여넣기 활동  
- AI 요청량 급증 시 경고 알림  
  
4) 사용자 행동 기반 탐지  
- AI 활용 목적 입력 분석 (업무 관련/비업무 구분)  
- 민감 키워드 포함 여부 기반 분류 및 리포트

## 3.3 시스템 구성 예시

- Shadow AI 탐지 시스템 구성 요소:  
 • 네트워크 흐름 감지 모듈 (NetFlow 기반)  
 • 브라우저/OS 수준 Agent  
 • 로그 수집기 (Syslog, Winlogbeat 등)  
 • AI 탐지 시각화 대시보드 (Grafana/ELK 등)

## 3.4 대응 프로세스

1단계: 비인가 AI 사용 탐지 →  
2단계: 사용자 및 부서에 알림 전송 →  
3단계: 접근 차단 또는 사용 정지 →  
4단계: 보안팀 보고 및 교육 재이수 조치

# 4. AI 사용 로그 대시보드 구성

## 4.1 목적

- AI 사용 내역을 시각화하고 이상 행위 및 트렌드를 분석함으로써 정책 이행 및 보안 수준을 점검하고 개선

## 4.2 수집 항목

- 사용자 정보: 이름, 사번, 부서  
- 접속 시간, IP, 디바이스 정보  
- AI 사용 플랫폼 및 모델명 (예: ChatGPT, Claude)  
- 입력/출력 텍스트 샘플 (요약 또는 일부 마스킹)  
- 탐지된 민감정보 항목  
- 정책 위반 여부 및 경고 이력

## 4.3 기술 구성

- 로그 수집기: Filebeat, Winlogbeat, syslog 등  
- 데이터 저장소: Elasticsearch, InfluxDB 등  
- 시각화 도구: Kibana, Grafana, Metabase  
- 관리자 기능: 필터링, 사용자별 로그 조회, 알림 기능

# 5. 내부 LLM 도입 방안

## 5.1 목적 및 기대효과

- 외부 API 의존도 감소 및 데이터 주권 확보  
- 사내 지식 기반과 연계된 맞춤형 AI 제공  
- 비용 효율성 및 보안성 향상

## 5.2 구축 전략

1) 모델 선정  
- 경량 오픈소스 LLM 우선 (예: LLaMA2, Mistral, OpenChat)  
- 지속 가능성과 커뮤니티 기반 고려  
  
2) 학습 및 파인튜닝  
- 사내 문서 기반 Embedding 및 Retrieval  
- Retrieval-Augmented Generation (RAG) 구조 적용  
  
3) 인프라 구성  
- On-Prem 서버 또는 Private Cloud  
- GPU 서버 또는 가상화 기반 클러스터 (Kubernetes)  
  
4) 인터페이스  
- 내부 API 또는 웹 기반 챗봇 제공

# 6. ISMS-P 기반 AI 보안 감사 항목 설계

## 6.1 필요성

- AI 사용이 정보보호 관리체계(ISMS-P)의 통제 항목에 포함되지 않으면 인증 유지 및 규정 위반 가능성 존재

## 6.2 감사 항목 예시

- AI 사용 목적 및 범위 정의 여부  
- AI 사용 정책 수립 및 공지 여부  
- 사용 로그 보관 및 감사 가능 여부  
- 민감정보 자동 탐지 및 차단 시스템 존재 여부  
- 외부 AI 서비스에 대한 계약 및 보안 검토 여부  
- Shadow AI 탐지 체계 운영 여부  
- 내부 AI 솔루션의 보안 및 유지관리 체계

## 6.3 연계 통제 항목 (예시)

- ISMS-P 제6영역: 정보보호대책의 구현  
 • 6.3.2 외부 위탁 관리  
 • 6.4.3 정보시스템 운영 보안  
 • 6.4.5 로그 및 모니터링  
 • 6.5.3 민감정보 및 중요정보 보호 조치

# 7. 브랜딩 전략 및 명칭 (잠정 확정)

## 7.1 메인 브랜드명

- 브랜드명: KRA-AIGov  
- 의미: Krase사의 AI 거버넌스 및 보안 통제 프레임워크를 상징하는 명칭  
- 목적: 기업 내 AI 사용 통제, Shadow AI 탐지, 내부 LLM 운영, AI 감사 등 통합 관리

## 7.2 부속 시스템 명칭 (잠정)

1) PromptShield  
 - 역할: 프롬프트 입력 필터링 및 외부 API 프록시 제어  
 - 유의사항: 유사 사용 사례 존재, 상표 등록 필요  
  
2) Shadoweye  
 - 역할: Shadow AI 및 비인가 AI 사용 탐지 시스템  
 - 유의사항: 유사 상표 존재 가능성 있음, 상표 등록 권장  
  
3) DashAILog  
 - 역할: AI 사용 로그 수집 및 시각화 대시보드  
 - 유의사항: 현재로서는 상표 등록 이슈 없음  
  
4) FortLLM  
 - 역할: 내부 LLM 제공 및 사내 전용 AI 응답 시스템  
 - 유의사항: 현재로서는 상표 등록 이슈 없음

## 7.3 권장 조치 및 향후 방향

- 모든 명칭에 대해 상표 등록 검토 및 법률 자문 진행 필요  
- 사내 포스터, 인트라넷, 보안 공지 등에 일관되게 브랜딩 적용  
- 브랜드 컬러, 로고, 슬로건 등 시각 정체성(BI) 구성 예정

# 8. KRA-AIGov 개발을 위한 오픈소스 구성

## 8.1 기능별 오픈소스 도구

KRA-AIGov 솔루션 개발을 위해 활용 가능한 오픈소스 솔루션은 다음과 같이 기능별로 정리됩니다.

### 1) 입력/출력 필터링 및 정책 게이트웨이

- OpenLLM Guard: LLM 응답 필터링 및 제어  
- Rebuff: Prompt injection 방지 및 프록시 형태로 사용  
- LangChain Guardrails: 프롬프트 제한 규칙 적용  
- FastAPI + Regex NLP 필터: 커스텀 프록시 서버 구현  
- Pangea Redact: 개인정보 자동 마스킹 API

### 2) Shadow AI 탐지 / 행위 분석

- Zeek: AI API 접속 추적용 네트워크 분석 도구  
- Wazuh: SIEM/EDR 통합 보안 플랫폼  
- Elastic Agent + Kibana: 로그 수집 및 시각화  
- OSQuery: 로컬 앱 설치 및 상태 모니터링

### 3) AI 사용 로그 및 대시보드

- ELK Stack (Elasticsearch, Logstash, Kibana): 로그 수집/저장/시각화  
- Grafana + Loki: 실시간 로그 모니터링  
- Prometheus + Alertmanager: 이벤트 기반 경고 시스템

### 4) 내부 LLM 운용 및 제어

- Ollama: LLaMA, Mistral 등 경량 LLM 실행  
- LangChain / LlamaIndex: 문서 기반 RAG 시스템  
- PrivateGPT: 로컬 GPT 환경  
- RAGflow: 오픈소스 기반 RAG 파이프라인

### 5) 정책 엔진 및 인증/인가 관리

- OPA (Open Policy Agent): 정책 기반 접근 제어  
- Keycloak: 인증/인가 및 SSO  
- Vault (Hashicorp): AI API Key 및 민감정보 보안 저장소

## 8.2 오픈소스 기반 시스템 설계 예시

[KRA-AIGov]  
 ├─ PromptShield : Rebuff + Guardrails + FastAPI  
 ├─ Shadoweye : Zeek + OSQuery + Wazuh + Kibana  
 ├─ DashAILog : ELK Stack + Grafana + Alertmanager  
 └─ FortLLM : Ollama + PrivateGPT + LlamaIndex  
 ▲  
 인증/정책 : Keycloak + OPA + Vault

# 9. MVP 기능 정의 및 개발 WBS

## 9.1 추천 MVP 기능

KRA-AIGov MVP는 핵심 보안 기능을 작고 빠르게 검증하기 위한 경량 버전으로, 다음의 기능을 포함합니다:  
1) AI Proxy Gateway (PromptShield)  
2) AI 사용 로그 수집 및 시각화 (DashAILog)  
3) Shadow AI 탐지 기능 (Shadoweye)  
4) 민감 정보 마스킹  
5) 기본 정책 관리 인터페이스 (Web UI 또는 config 기반)

## 9.2 개발 WBS (Work Breakdown Structure)

### 1. 프로젝트 환경 구성

* - 1.1 개발 환경 설정 (Docker, Git, Python venv)
* - 1.2 오픈소스 도구 선정 및 구조 설계
* - 1.3 서버/테스트 PC 준비 및 Agent 설치

### 2. AI Proxy Gateway 개발

* - 2.1 FastAPI 프록시 서버 개발
* - 2.2 입력 필터링 기능 구현 (정규식 + 키워드 DB)
* - 2.3 출력 응답 필터링 및 로깅 연동
* - 2.4 예외 처리 및 테스트

### 3. 로그 수집 및 시각화

* - 3.1 Logstash 설정 및 수집 파이프라인 구성
* - 3.2 Elasticsearch 저장소 구축
* - 3.3 Kibana 대시보드 생성 (사용자별/시간별/유형별 필터)

### 4. Shadow AI 탐지 기능

* - 4.1 OSQuery 또는 Wazuh Agent 설치
* - 4.2 비인가 AI 사용 탐지 룰 작성
* - 4.3 탐지 시 알림 기능 연동 (Slack/email 등)

### 5. 민감 정보 마스킹 기능

* - 5.1 PII 키워드 목록 정의
* - 5.2 마스킹 엔진 개발 (정규표현식 or Presidio 연동)
* - 5.3 입력/응답 마스킹 적용 및 테스트

### 6. 정책 관리 인터페이스

* - 6.1 기본 정책 파일(YAML/JSON) 구조 설계
* - 6.2 Flask/Django 기반 Admin Web UI 제작 (선택)
* - 6.3 정책 변경 시 반영 자동화 로직

### 7. 통합 테스트 및 PoC 적용

* - 7.1 기능 통합 및 시나리오별 테스트 수행
* - 7.2 시범 도입 부서 선정 및 설치
* - 7.3 운영 리포트 수집 및 피드백 반영

# 10. VDI 환경 고려사항 및 Agentless 기반 MVP 전략

## 10.1 VDI 환경의 특성

VDI(Virtual Desktop Infrastructure) 환경은 가상 데스크탑이 사용자 로그인 시 동적으로 생성되는 구조로,  
전통적인 사용자 단말기 기반 보안 에이전트 설치 방식과는 다른 접근이 필요합니다.  
  
특히, 비영속성(Non-persistent) VDI에서는 에이전트가 매 세션마다 초기화되며, 리소스 및 유지보수 이슈가 존재합니다.  
이에 따라, KRA-AIGov의 MVP는 VDI 환경에 적합한 Agentless(에이전트 설치 없이) 방식으로 설계합니다.

## 10.2 권장 아키텍처 구성

1) AI 사용은 모두 중앙의 Proxy Gateway를 통해 이루어지도록 설정  
2) 모든 요청/응답은 서버에서 필터링, 로깅, 정책 적용  
3) VDI 사용자 ID, 부서 정보를 기반으로 정책 적용  
4) 접속 기록은 VDI 포탈 로그 또는 중앙 로그 수집기를 통해 확인

## 10.3 기술 구성 요소

- Proxy Gateway: FastAPI 기반, PromptShield 역할 수행  
- 민감정보 마스킹: 서버 측 NLP 또는 정규표현식 기반 필터링  
- 로그 수집 및 대시보드: ELK Stack (ElasticSearch + Kibana)  
- 정책 적용: OPA (Open Policy Agent) + YAML 정책 파일  
- VDI 접속 연동: Horizon/Netscaler 로그 또는 SSO 연계 기반 사용자 식별

## 10.4 MVP 주요 기능 흐름 (Agentless)

1) 사용자는 사내 VDI를 통해 인터넷에 접속  
2) 모든 AI 서비스 요청은 Proxy Gateway를 통과  
3) 민감정보 탐지 및 정책에 따라 요청 차단 또는 마스킹  
4) 사용 로그는 실시간 저장 및 Kibana 대시보드로 시각화  
5) 관리자 Web UI에서 사용자별 로그 조회 및 정책 수정 가능

## 10.5 기대 효과

- 사용자 PC 또는 가상 데스크탑에 별도 Agent 설치 불필요  
- 모든 트래픽은 중앙 서버에서 통제되어 보안성 향상  
- 비인가 AI 사용 차단 및 기록 가능  
- Shadow AI 또는 의심 행위 탐지는 네트워크 수준에서 대응 가능

# 11. VDI 환경 Agentless 방식의 AI 보안솔루션 MVP 개발 WBS 상세화

## 1. 프로젝트 준비

* - 1.1 VDI 환경 구조 분석 (Horizon, Citrix, Netscaler 등)
* - 1.2 네트워크 흐름 파악 및 트래픽 통제 지점 정의
* - 1.3 요구사항 기반 MVP 범위 명확화 및 리스크 정리

## 2. Proxy Gateway 구축

* - 2.1 FastAPI 또는 Nginx 기반 프록시 서버 구축
* - 2.2 Prompt 필터링 로직 구현 (정규식 + 키워드 DB)
* - 2.3 응답 필터링 및 마스킹 로직 내장
* - 2.4 프록시 서버 SSL 인증서 적용 및 보안 설정

## 3. 로그 수집 및 분석

* - 3.1 프록시 로그 → Filebeat로 Logstash 연동
* - 3.2 Elasticsearch 저장소 구성
* - 3.3 Kibana 대시보드 설계 (사용자, 시간, 위반 키워드 필터)
* - 3.4 사용자 ID 매핑 구조 설정 (VDI 접속 정보 연동)

## 4. 정책 적용 엔진 설계

* - 4.1 Open Policy Agent (OPA) 구성
* - 4.2 사용자/부서별 YAML 정책 구조 설계
* - 4.3 정책 적용 시 프록시 서버 연동 방식 구현
* - 4.4 정책 수정 시 핫리로드 또는 CI/CD 연동

## 5. 민감정보 마스킹 모듈

* - 5.1 주민번호, 전화번호, 이메일 등 탐지 정규표현식 구축
* - 5.2 NLP 기반 PII 탐지 라이브러리 테스트 (예: Presidio)
* - 5.3 실시간 필터링/마스킹 테스트 및 로그 저장 여부 확인

## 6. VDI 포탈 연계 및 사용자 인증 매핑

* - 6.1 Horizon/Netscaler 사용자 접속 로그 수집 설정
* - 6.2 접속 로그 기반 사용자 ID 추출 방식 확정
* - 6.3 사용자별 AI 사용 로그와 정책 연계 검증

## 7. 통합 테스트 및 시범 적용

* - 7.1 각 모듈 통합 및 흐름 테스트 (입력 → 필터 → 로깅)
* - 7.2 시범 부서 적용 및 이상행위 탐지 테스트
* - 7.3 관리자 피드백 반영 및 정책 튜닝 반복

## 11.1 VDI 환경 구조 분석 (전문가 수준 상세 설명)

본 단계는 KRA-AIGov를 VDI 환경에 효과적으로 통합하기 위해 반드시 수행되어야 하는 초기 분석 단계로,   
VDI 아키텍처와 트래픽 흐름, 사용자 인증 연계 가능성, 로그 수집 위치 등을 전문가 수준에서 파악하는 것을 목표로 합니다.

### 1) VDI 시스템 구성 요소 분석

- Connection Broker: VMware Horizon, Citrix Controller 등의 세션 분배자  
- VDI Desktop Pool: 사용자의 가상 데스크탑 이미지 (영속/비영속 여부 분석)  
- VDI Gateway: 외부 접속 관문 (Netscaler, VMware UAG 등)  
- Management Console: 정책 설정 및 사용자 정보 연계  
- Authentication Server: LDAP, AD, SSO 로그인 연동 여부  
- Network Layer: 외부 인터넷으로 나가는 트래픽 경로 파악 (프록시/방화벽/NAT)

### 2) 트래픽 흐름 분석 (AI 요청 관점)

- 사용자가 포탈 로그인 후 가상 데스크탑에 접속  
- AI 사용 시 요청은 VDI 내부 네트워크 → Virtual Switch → NAT 또는 Proxy → 외부로 전송  
- AI 응답은 동일 경로로 되돌아옴  
  
✔️ 분석 포인트:  
- 프록시 삽입 가능 지점  
- 사용자 ID와 요청 IP/세션 매핑 가능 여부

### 3) 로그 수집 지점 및 도구

- VDI Gateway: 세션 로그 (Citrix ADM, Horizon Logs)  
- VDI OS: 앱 실행 로그, 도메인 접속 로그 (Windows Security Log)  
- 프록시 서버: AI 사용 기록 (FastAPI/Nginx 로그)  
- 통합 SIEM: ELK, Wazuh, Splunk 등 연계 가능

### 4) 분석 결과에 포함되어야 할 내용

- VDI 접속 경로 및 네트워크 구조 다이어그램  
- AI 서비스 사용 시 트래픽 흐름 시퀀스  
- Shadow AI 감지 불가능 영역 정의  
- 인증정보 추출 가능성 및 로그 수집 가능 위치

### 5) 보안 및 정책 고려 사항

- LDAP/AD 사용자 그룹 기반 정책 연계 설계  
- TLS 가시성 확보 (프록시에서 복호화 필요 여부)  
- AI 사용 시 프록시 우회 불가능하도록 방화벽 정책 필요

### ✔️ 전문가 체크리스트

- [ ] VDI 포탈 종류 및 로그인 방식 파악 (SSO 포함)  
- [ ] NAT/프록시 경유 여부 확인  
- [ ] 사용자 ID 추적 가능한 로그 위치 확보  
- [ ] Shadow AI 발생 가능 경로 식별  
- [ ] 프록시 삽입 가능 지점 확인  
- [ ] 로그 수집 포맷 및 도구 정리 (ELK, Wazuh 등)

## 11.3 요구사항 기반 MVP 범위 명확화 및 리스크 정리 (전문가 수준 상세 설명)

이 단계는 KRA-AIGov의 성공적인 개발과 PoC 실행을 위해 MVP의 기능 범위를 명확히 정의하고,   
이해관계자의 요구사항을 반영하며, 예상 리스크를 식별하고 대응책을 수립하는 것을 목표로 합니다.

### 1) 이해관계자 요구사항 정리

- 보안팀: 외부 AI 사용 차단, 로그 확보, 민감정보 검출, ISMS-P 대응  
- IT운영팀: 시스템 경량화, Agentless, 로그 통합  
- 사용자부서: 최소한의 사용 허용, 불필요한 감시 제거  
- 경영진: Shadow AI 리스크 수치화, ROI 입증 가능

### 2) MVP 기능 범위 기준

[포함]  
- AI Proxy Gateway (입력/응답 필터링)  
- 로그 수집 및 Kibana 시각화  
- Shadow AI 탐지 (Agentless)  
- 민감정보 마스킹  
- 정책 파일(YAML) 기반 정책 적용  
  
[제외]  
- Agent 기반 사용자 행위 추적  
- FortLLM 등 내부 LLM 응답 시스템  
- 고급 문맥기반 응답 제어  
- 관리자 권한 UI, 알림 시스템 등 운영 기능

### 3) 예상 리스크 및 대응 전략

- 기술적 리스크: 프록시 필터링 성능 저하 → 경량화 + SSL Termination 검토  
- 로그 식별 문제: 사용자 ID 불명확 → 포탈 로그/세션ID 연계  
- 오탐지/과차단: 정상 업무 필터링 → 화이트리스트 반영, 부서별 튜닝  
- 사용자 반발: 감시 인식 → 교육/사전고지, 피드백 채널 운영  
- Shadow AI 우회: DNS/IP 직접접근 → DNS 기반 차단 정책 보완

### 4) 주요 산출물

- 이해관계자 요구사항 정리 문서  
- 포함/제외 기능 리스트 (MVP 정의서)  
- 리스크 항목 및 대응 전략표  
- 기능 우선순위 문서 또는 일정표