RedOpsAI => ROA

v1.0.0

RedOpsAi : Red-team Operation AI

- 컨셉

채팅을 통한 원격지 서버의 위협 식별 및 조치

- 요약 플로우

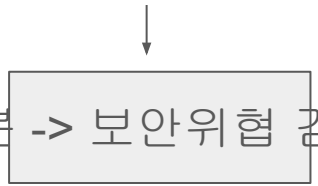
1. 채팅 -> 대상 설정 -> 열린 포트 스캔 -> 포트와 서비스 매칭 -> 서비스의 CVE 취약점 분석 -> 결과 수집
2. 웹서버일 경우 -> 웹 취약점 점검 전용 에이전트 수행 -> 결과 수집
3. 1, 2 번에서 수집된 결과로 보고서 생성

보안 컨설팅 방법론 개편

기존

환경분석(자산식별) -> 취약점 진단 -> 위험평가 -> 문서화(보호대책 수립)

개편안



환경분석(자산식별) -> 취약점 진단, 해당 취약부분 -> 보안위협 검증 -> 위험평가 -> 문서

* 모의해킹 : 자산의 실제 위험에 따른 구체적인 평가

모의해킹 프로세스

정보 수집-> 분석-> 공격 가능한 루트 선택 -> 실제 침투 시나리오 수행 -> 결과 보고서 생성 -> 이에 따른 조치 가이드 또는 ai agent로 조치 수행 -> 이행 점검

PoC 모델에서는 CVE 와 웹 취약점을 우선 과제로 진행

프론트 기준 및 사용할 오픈 소스

1. 벡터 DB 사용 가능 여부(오픈 소스 해당)
2. 개발 언어
3. RAG 기능 지원 여부
4. 랭그래프와 랭체인 연동 가능 여부
5. 여러 벤더들과 커스텀 llm api 연동 가능 여부
6. mcp 연동 가능여부
7. react 사용

후보 목록

- **Dify(production)**
- **AnythingLLM(PoC)**
- **MaxKB**
- **LibreChat**
- **Chatbot UI / Lobe Chat**
- **open-webui(PoC)**

프론트 open-webui 기준

- 추가 기능 개선 항목

구글 로그인 기능 추가

크레딧 개념 추가

감사 로깅 필요

점검 결과

SIEM 연동 // 기 구축된 업체와의 연계계

시스템 설정 <- ex. 인증서 적용, 모델 사용

- 추가 연동 항목

1. 무료 웹 검색 엔진

SearXNG : fork 해서 작업 커스텀 필요(latest 오류)

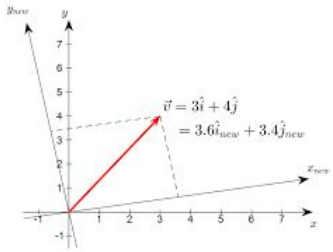
대안 : https://github.com/benbusby/whoogle-search

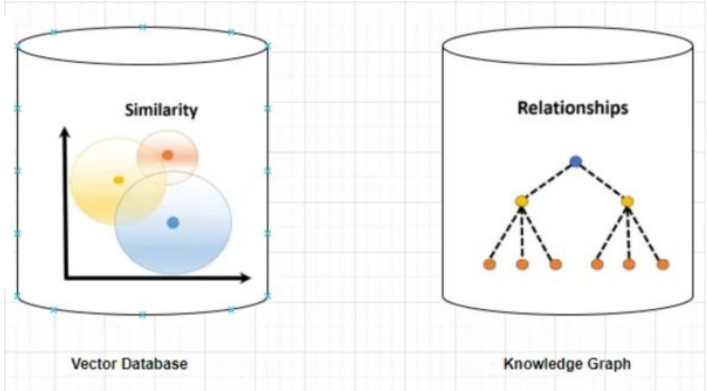
2. MCP : https://github.com/0x4m4/hexstrike-ai

3. Dify : flow 벤치마킹 => 플로우 편집하는 기능

백엔드 기준(UI 중복 기능

1. RDS, 벡터 DB 사용
2. 파이썬, 노드 사용
3. MCP 연동(피싱메일, 모의해킹 템플릿)
4. 별도 구축된 올라마 연동
5. chatgpt api 우선 연동 -> 과금이슈





node

python™

docker

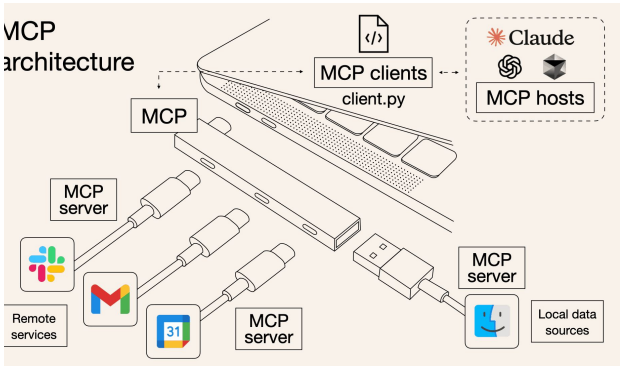
ChatGPT

Get up and running with large language models.

Run Llama 3.3, DeepSeek-R1, Phi-4, Mistral, Gemma 2, and other models, locally.

Download ↓

Available for macOS, Linux, and Windows



에이전트 개별 예시(리서치 부분)

domain2ip : 도메인 기반에서 추출되는 대상을 식별(Reverse DNS 등)

scanning: nmap 열린 포트 수집

service agent : 열린 포트에 대한 서비스 정보 수집

threat agent : 서비스의 버전이나 정보들로 위협(exploit) 이 가능한 정보가 있는지 검색 및 시나리오 목록화, 연계 분석이 핵심 CoT 를 해야 함

attack agent : 식별된 위협으로 실제 침투 테스트를 수행하는 에이전트

feeadback agent : 침투 테스트 결과를 수집하여 보고서 형태의 피드백 에이전트

config agent : 사용자가 조치를 위해서 설치하는 에이전트로 피드백 기반의 개선 사항을 반영하는 에이전트

- 클라우드에서 사용하기 어려울 듯? / 방화벽 있으면 포트 스캐닝 자체가 가능한가?

주요 기능(Production)

- 작업
  - 채팅
    - ai 와 수동으로 보안 점검 수행
  - 플로우
    - 템플릿 관리 및 실행
    - 스케줄 관리

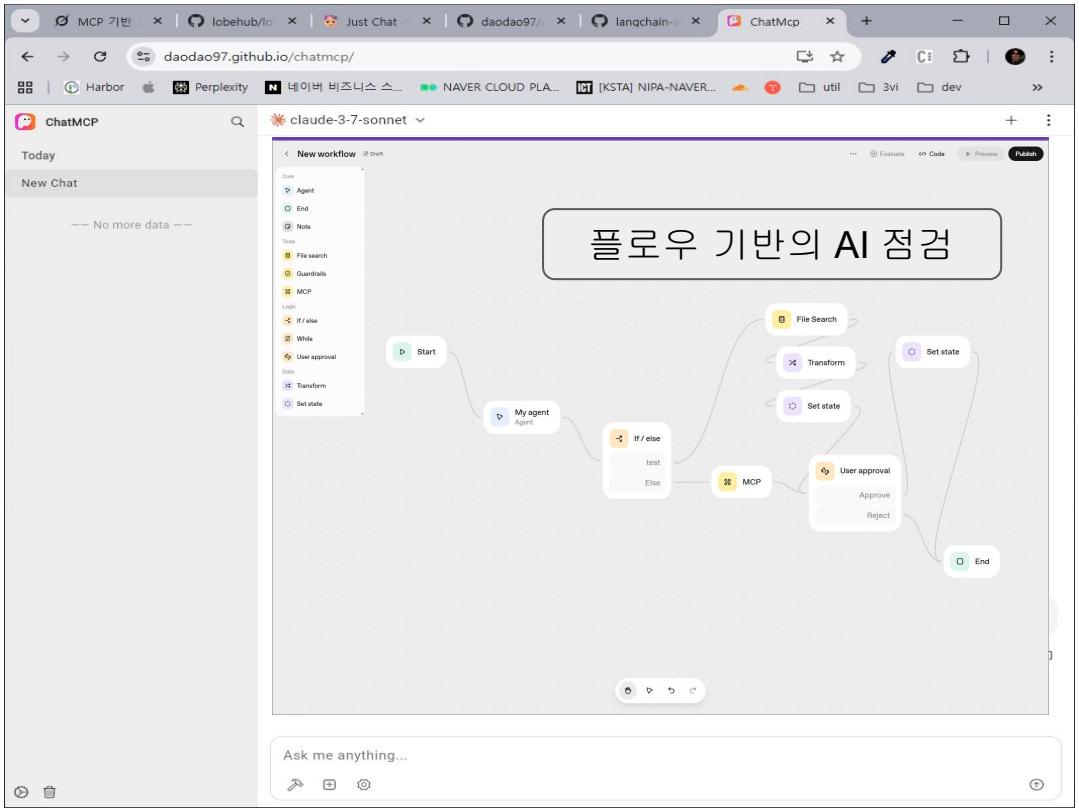
- 결과
  - 점검 결과
    - 침투 시나리오
    - 위협 및 조치 가이드
    - 재점검

- 모듈 관리
  - MCP
    - 기본 : 기본 제공
    - 확장 : 사용자가 직접 등록해서 사용 MCP 프로토콜 사용
  - Agent
    - AI Agent
    - 공용 설정(timeout 등)

- 과금
  - llm 토큰 기반 사용량 계측
  - 올라마 셀프 구축일 경우 해당 없음
- 기업
  - 사용자별 설정
  - 기업 설정
  - SEIM 연동 설정
    - hook
    - syslog
- 로그
  - 감사로그
  - 시스템 알림(리소스 및 시스템 오류)
  -
- 시스템 설정
  - license
  - 알림
    - SMTP
    - webhook
    - syslog
  - 회원가입 여부 -> 관리자 패널에 존재
  - 구글 로그인 사용 여부 -> 관리자 패널에 추

화면 구성 - 모의해킹

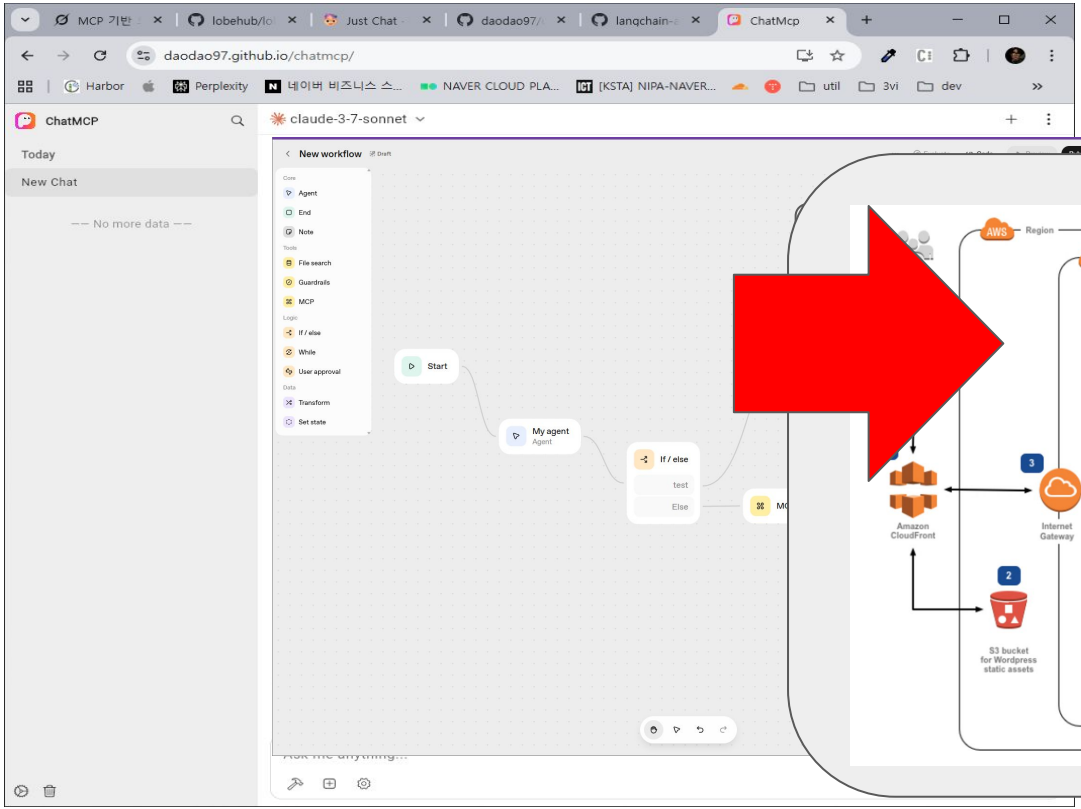
ai 템플릿 생성 -> 모의해킹 템플릿(커스텀 가) -> 실행



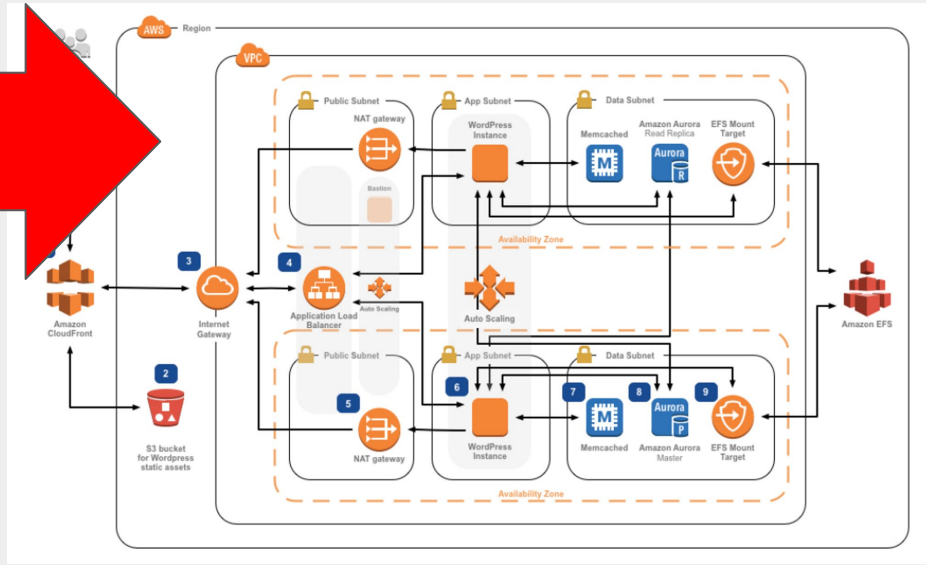
2025 openai dev day : https://openai.com/index/introducing-agentkit/

화면 구성 - 내부망 자산식별

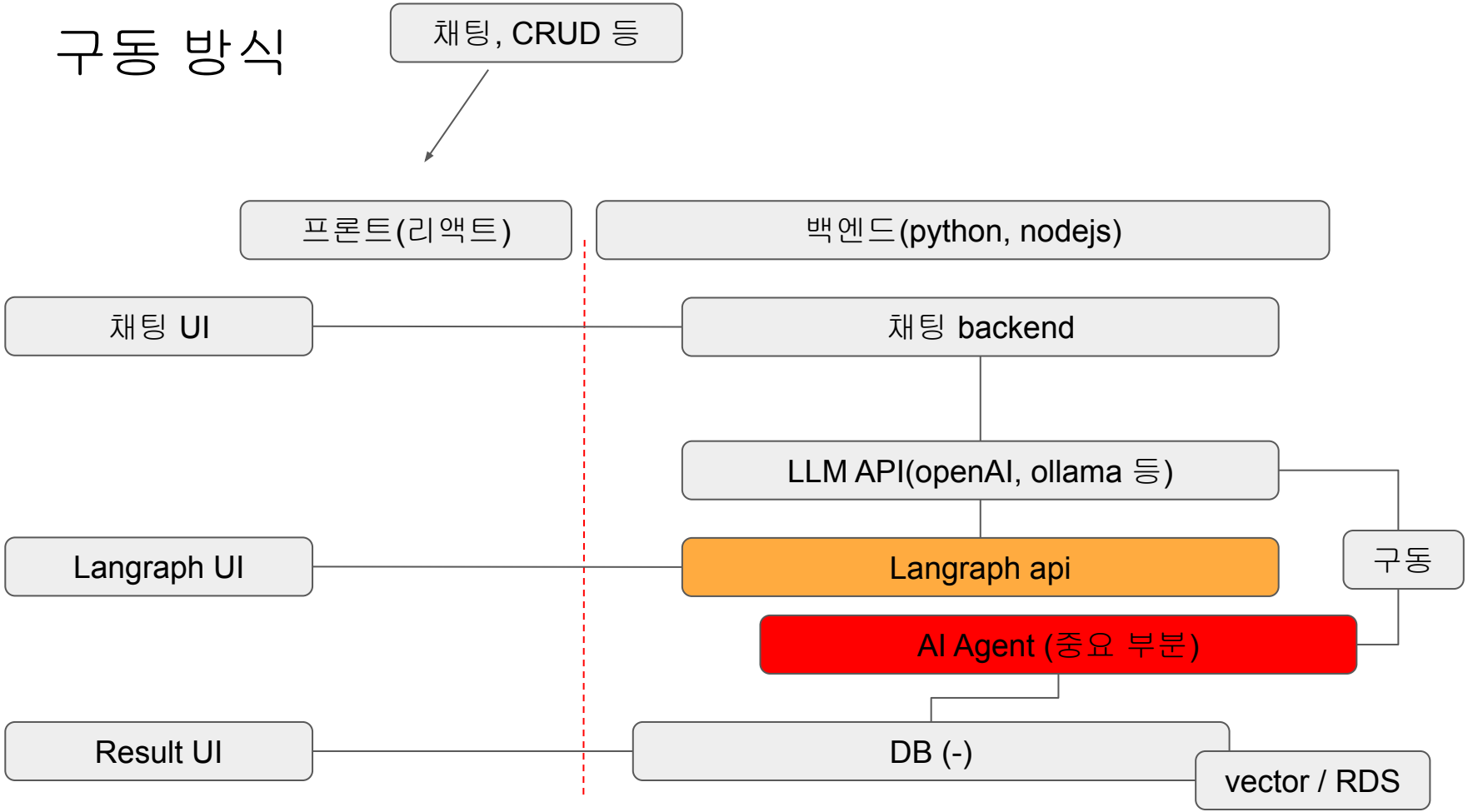
> netdiscover -r [네트워크 대역]
> nmap -sT [네트워크 대역]



식별된 자산

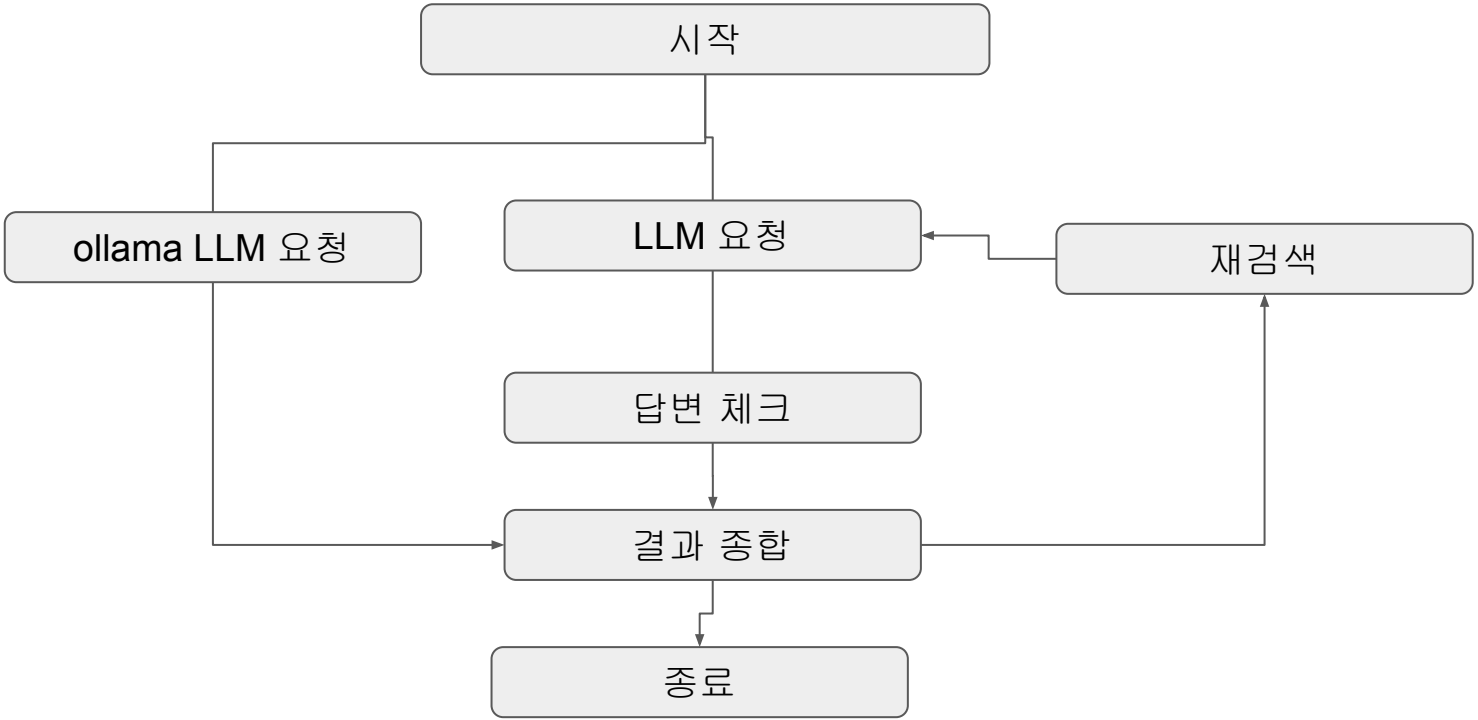


구동 방식


채팅, CRUD 등
프론트(리액트)
백엔드(python, nodejs)
채팅 UI
채팅 backend
LLM API(openAI, ollama 등)
Langraph UI
Langraph api
구동
AI Agent (중요 부분)
Result UI
DB (-)
vector / RDS

LangGraph(Chain of Thought 필요) 활용 필

학습 필요 : https://github.com/teddylee777/langchain-kr



그외

AI 모델 => 과금이슈? => 올라마 사용

내부 모델 // 보안 데이터가 내부 서버에서 로직을 거치는가?

모듈 관리가 핵심

맞춤형 MCP + CWE 취약점 진단(스크립트)까지 포함되어야 함