Prowler MCP 개발 및 AWS 취약점 점검 / 조치 자동화 아키텍처 구성기

김서연
AWS BEGINNER

2025. 8. 21

PROWLER MCP 제작기

Agenda

1. INTRO

- 최근 규제 변화
- CSPM 도구의 필요성
- WHAT IS PROWLER?
- PROLWER V5

2. MCP 제작

- 기본 아이디어
- CLAUDE DESKTOP MCP 제작
- 시연영상

3. CLOSING

- 향후 계획

1. INTRO

1.1 최근 규제 변화

: 금융 산업 내 AI 및 SaaS(클라우드) 사용 규제 완화

금융권 클라우드 전환 가속화

❷ 클라우드 이용 확대

(규제샌드박스)

	현 행	개 선
데이터	개인신용정보 금지	**가명정보 허용**
프로그램 유형	협업툴, 인사관리 등 비중요업무 허용	**고객관리(CRM), 업무자동화 등 추가 허용**
단말기	유선 PC만 허용	**모바일단말 허용**

2024년, 금융 산업 내 AI 및 SaaS(클라우드) 사용을 완화
(금융분야 망분리 개선 로드맵)

하나은행, 토스증권 등 시중 은행에서의
AWS(클라우드) 사용 증대

Ⅳ. 클라우드 활성화를 위한 제도개선 방안

〈 기본 방향 〉

1. 클라우드 서비스 이용범위 확대

□ 금융회사, 핀테크기업이 클라우드를 활용하여 **혁신적 상품과 서비스 개발**이 가능하도록 **이용범위를 확대**

○ **개인신용정보·고유식별정보**도 국내소재 **클라우드를 이용**할 수 있도록 개선

※ **국외소재 클라우드** 허용은 국내소재 클라우드 운영 이후 문제점 등을 고려하여 **중·장기적**으로 **검토**

2. 클라우드 서비스 이용·제공 기준 마련

□ 중요정보 처리시스템의 **안전성을 확보**하기 위해 **클라우드 이용**(금융회사), **제공**(제공자)**시 기준**을 도입하고 **운영방안**을 수립

○ **(금융회사) 중요정보** 클라우드 이용시 **안전성 관리**를 **강화**

○ **(제공자) 금융의 특수성을 반영**해 클라우드 서비스 제공자가 기본적으로 준수해야 할 **기준을 마련**

* '중요정보'의 경우 기존 금융권 전산시스템에 준하는 보안수준을 충족하도록 함

3. 클라우드 서비스 감독·검사 강화

□ 클라우드 활용 확대를 고려하여 금융권 클라우드 이용현황에 대한 **모니터링을 강화**하고, 적절한 **감독·검사 체계** 마련

○ 클라우드 서비스 이용 관련 금융회사의 **보고의무 강화**

○ 전자금융보조업자(클라우드 서비스 제공자)에 대한 감독당국의 **직접 감독·조사권**을 확보하는 방안을 검토(법개정 사항)

1.1 최근 규제 변화

: 금융 산업 내 AI 및 SaaS(클라우드) 사용 규제 완화

◆금융권 클라우드 도입률 92%…“전략적 자산으로 인식 변화”

조사 결과 국내 금융기관의 92%가 활용 수준과 관계없이 퍼블릭 클라우드를 도입한 상태다. 현재 사용하지 않으나 향후 3년 이내 도입을 계획 중인 기관까지 포함하면 96%에 달한다. 현재 사용하지도 않고 도입 계획도 없는 기관은 4%에 불과했다.

Security Engineer (클라우드 보안/분석)

토스플레이스 소속 | 정규직 | 초기 멤버

합류하게 될 팀에 대해 알려드려요

- 토스플레이스의 보안팀은 Security Engineer, Information Security Manager, Privacy Manager, IT Manager로 구성되어 있어요.
- Security팀은 안전한 토스플레이스 서비스를 만들기 위해 Infra팀, 제품(서비스)을 만드는 사일로, Legal팀 그리고 Compliance팀과 협업하고 있어요.
- 이외에도 Security팀의 각 직무 담당자는 계열사의 동일 직군 동료들과 활발한 교류를 통해 전문분야에 대한 고민을 나누며 협업해요.
- 토스플레이스에서는 Security팀의 초기멤버로서 안정적인 서비스를 제공할 수 있도록 보안체계를 만들어 나가는 경험을 할 수 있어요.

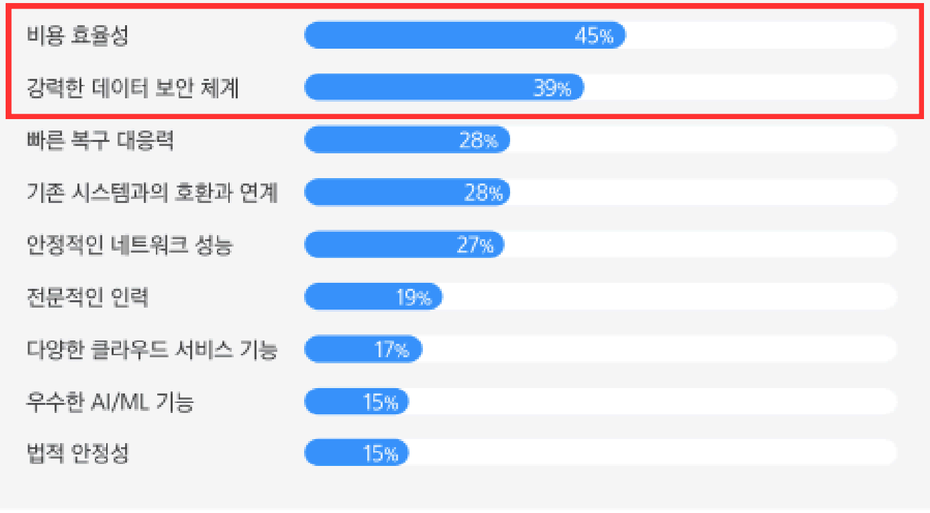
합류하면 함께할 업무예요

- AWS 보안 서비스를 운영하고 보안 인프라를 구축 및 관리해요.(FW, WAF, Shield, GuardDuty, IAM 등)
- AWS 취약점 점검 및 모니터링을 수행해요.(CSPM)
- AWS 인프라 취약점 점검(Server, DBMS, Network 등) 활동을 수행해요.
- 침해사고 분석 및 대응 업무를 수행해요.
- 다양한 보안 이벤트 분석을 통해 위협을 실시간으로 식별하고 대응해요.

1.2 CSPM 도구의 필요성

: 클라우드 도입 시 주요 요건인 보안 문제를 해결하기 위해 자동화 된 보안 점검 대응 시스템이 필요

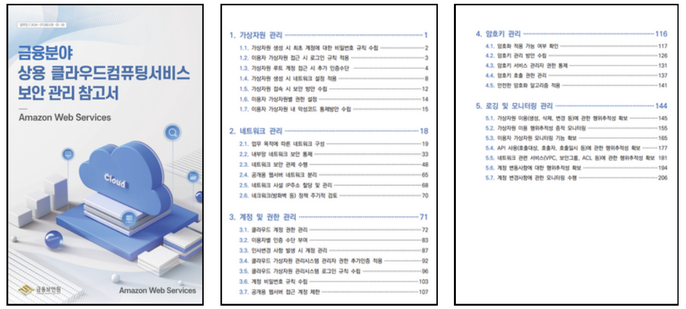
퍼블릭 클라우드 업체 선정 시 핵심 기능 요소



(산업별) Top 3
퍼블릭 클라우드 업체 선정 시 핵심 기능 요소

	제조	금융	리테일	서비스
1	비용 효율성	비용 효율성	비용 효율성	비용 효율성
2	강력한 데이터 보안 체계	강력한 데이터 보안 체계	강력한 데이터 보안 체계	강력한 데이터 보안 체계
3	빠른 복구 대응력	법적 안정성	빠른 복구 대응력	안정적인 네트워크 성능

자동화된 보안 점검 대응 시스템 필요



(금융)기업은 클라우드를 도입하면서 자산을 관리해야 하는데,
수작업으로 관리하면 시간과 인력이 과도하게 소요됨

자동화된 보안 점검 대응 시스템의 필요

1.2 CSPM 도구의 필요성

: 클라우드 도입 시 주요 요건인 보안 문제를 해결하기 위해 자동화 된 보안 점검 대응 시스템이 필요

1.3 WHAT IS PROWLER?

: 점유율이 가장 높은 오픈소스 CSPM 도구

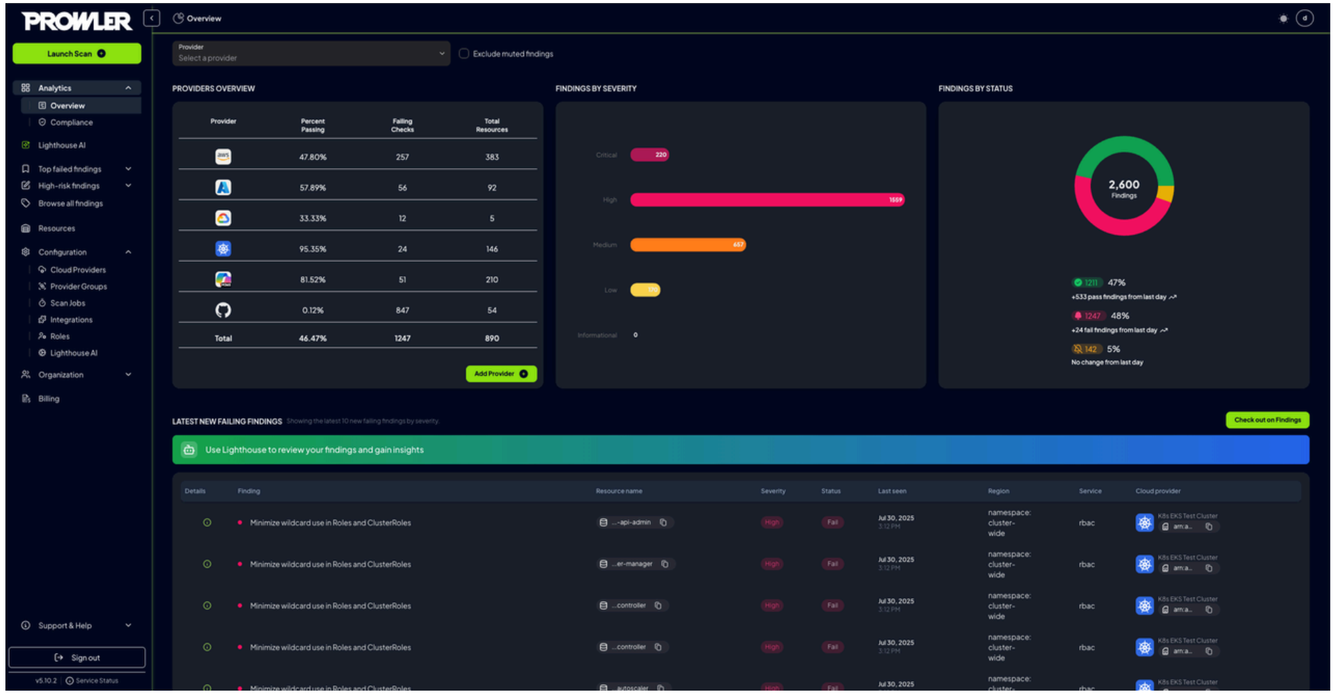
그림1-2. 주요 오픈소스 CSPM 포지셔닝

프로젝트명	GitHub Stars	GitHub 기여자 수	GitHub 최근 커밋 날짜	지원 플랫폼	지원 컴플라이언스	주요 기능
Prowler-cloud	10.9k	259	2024-11-20	AWS, Azure, Google Cloud, Kubernetes	KISA-ISMS-P,CIS, NIST 800-53, PCI-DSS, GDPR,FedRAMP,FFIEC,GXP,HIPAA, ISO 27001, SOC2, AWS Well-Architected Framework, ENS 등	클라우드보안 탐지 및 평가,자산 식별, 컴플라이언스 준수 점검, Incident Response 지원, 하드닝 및 포렌식 준비, 보고서 생성 ,AWS 시큐리티 허브 통합

1.3 WHAT IS PROWLER?

: 점유율이 가장 높은 오픈소스 CSPM 도구

Provider	Service	Status	Critical	High	Medium	Low	Muted
aws	accessanalyzer	FAIL (1)	0	0	0	1	0
aws	account	FAIL (1)	0	0	1	0	0
aws	lambda	FAIL (1)	0	0	0	1	5
aws	backup	FAIL (1)	0	0	0	1	0
aws	cloudformation	FAIL (5)	0	0	5	0	3
aws	cloudtrail	FAIL (4)	0	0	1	3	9
aws	cloudwatch	FAIL (19)	0	0	19	0	6
aws	config	PASS (1)	0	0	0	0	0

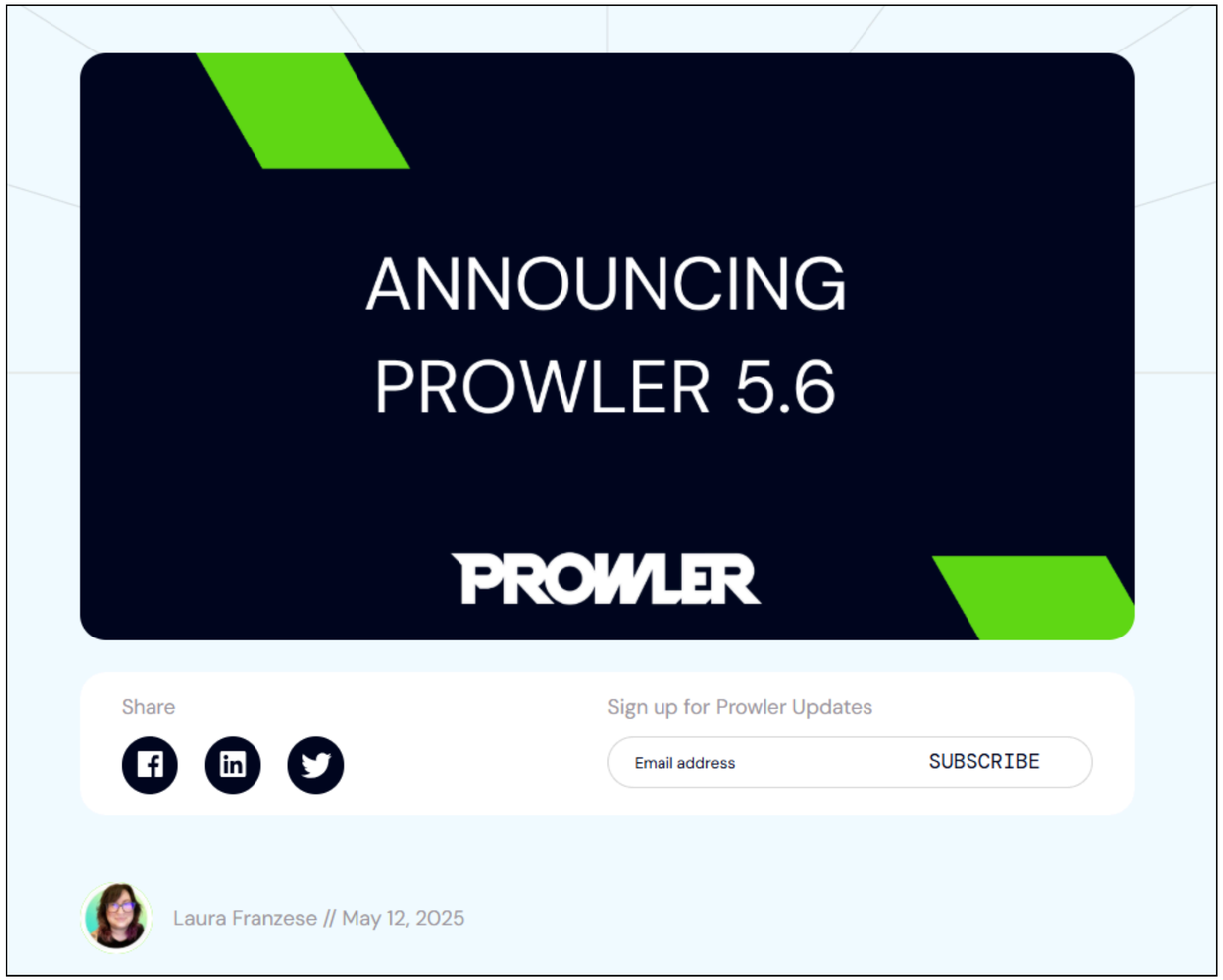


Prowler CLI

Prowler Cloud

1.4 PROWLER V5

: Prowler Studio를 필두로 한 AI 기능 확대 + MCP와 Fixer 기능 추가



About

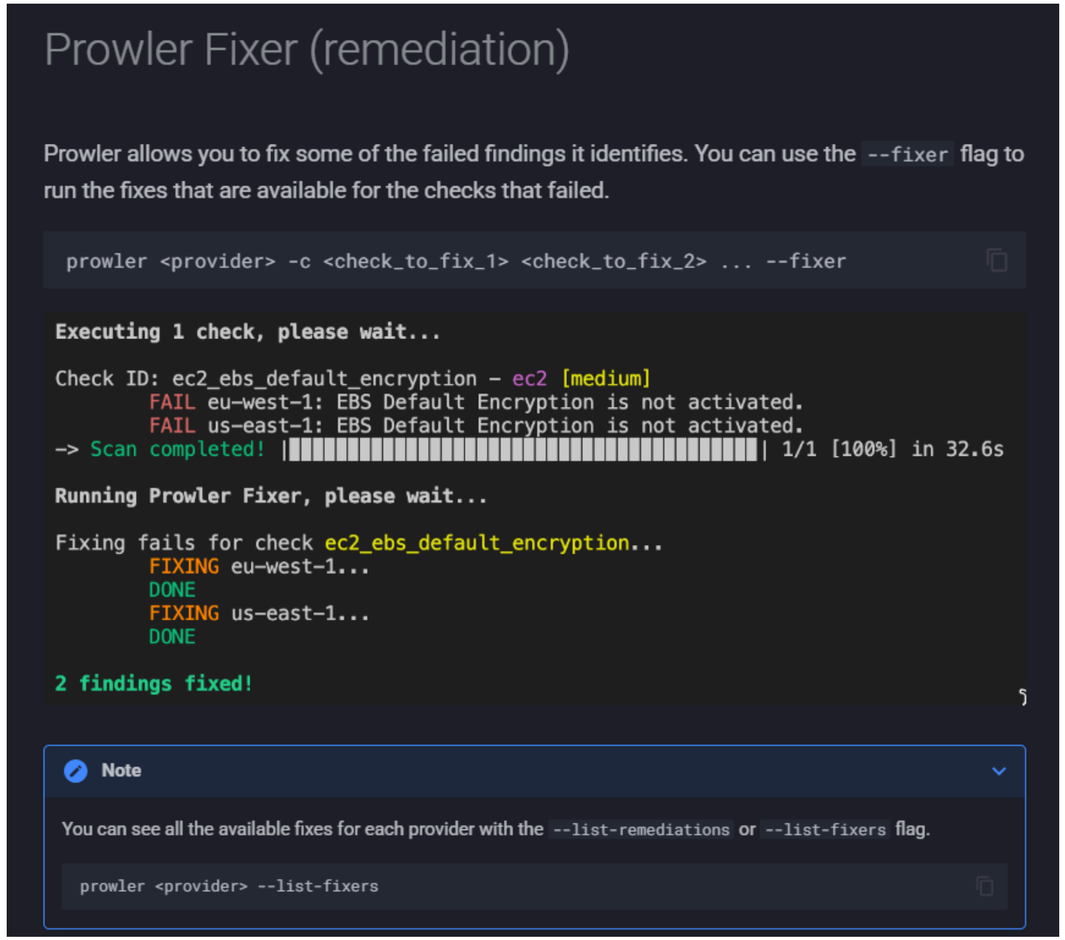
Prowler Studio is an AI assistant that helps you to create threat detection checks, remediations and update compliance frameworks for Prowler. It can be used as a CLI tool or as a web application.

security cloud threat controls compliance

AI 기능 강화와 자체 IaC 스니펫 추가

1.4 PROWLER V5

: Prowler Studio를 필두로 한 AI 기능 확대 + MCP와 Fixer 기능 추가



Prowler Fixer

보안 대응 IaC 스니펫

자동 수정 실행: --fixer 플래그를 사용해 진단 실패한 항목에 대한 자동 수정(Remediation) 명령을 실행

Fixer 코드 작성 가능: check_id_fixer.py 파일을 작성해, 특정 검사에 대해 리전 또는 리소스 단위로 해결 로직을 제공하는 Custom Fixer 구현이 가능

설정 기반 유연성 제공: --list-fixers 명령으로 사용 가능한 Fixer 목록을 확인할 수 있으며,
--fixer-config 옵션을 통해 YAML 기반 설정을 적용해 실행 동작을 커스터마이징 가능

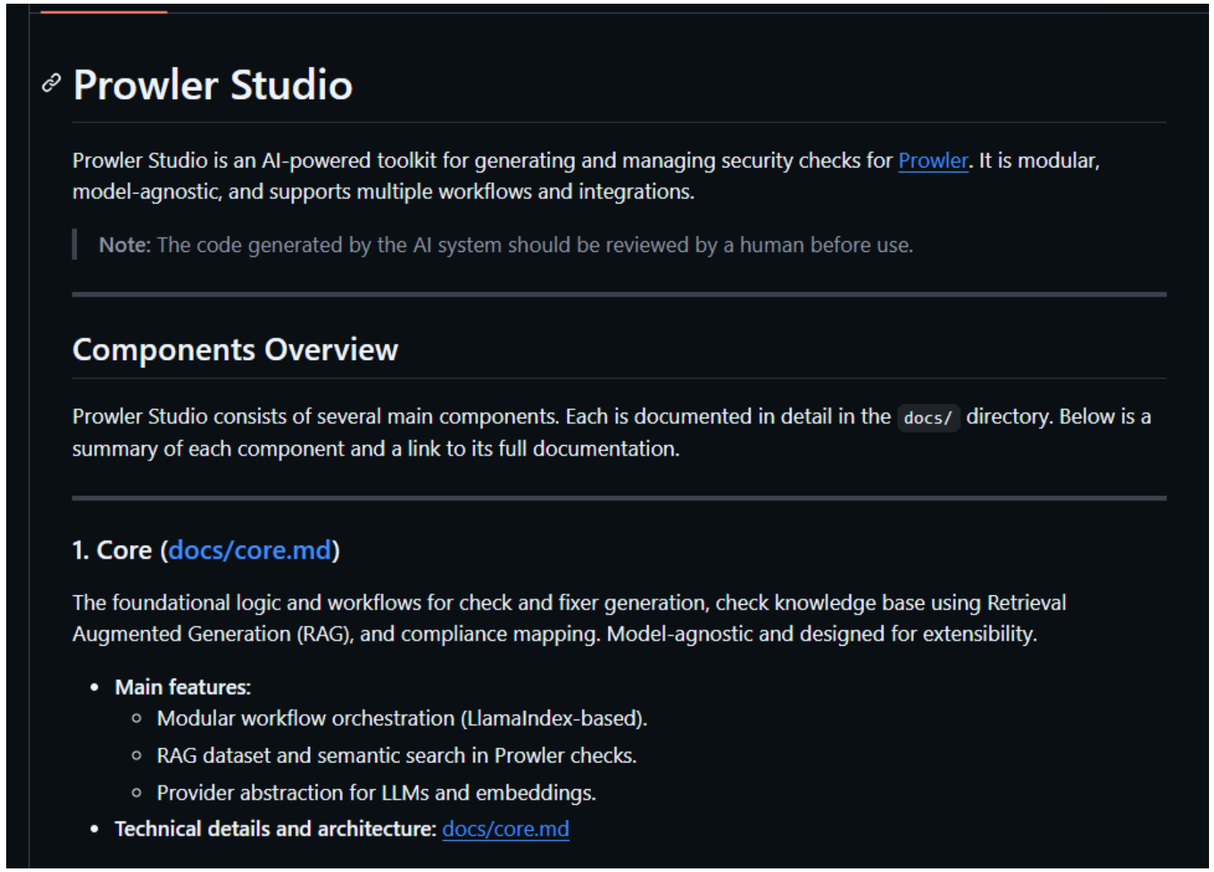
1.4 PROWLER V5

: Prowler Studio를 필두로 한 AI 기능 확대 + MCP와 Fixer 기능 추가


```
 _ __  _ __ _____      _| | ___ _ __
| '_ \| '__/ _ \ \ /\ / / |/ _ \ '__|
| |_) | | | (_) \ V  V /| |  __/ |
| .__/|_|  \___/ \_/\_/ |_|\___|_|v5.8.1
|_| the handy multi-cloud security tool

Date: 2025-08-21 16:30:33

- accessanalyzer_enabled
- awslambda_function_not_publicly_accessible
- cloudtrail_logs_s3_bucket_is_not_publicly_accessible
- cloudtrail_multi_region_enabled
- cloudtrail_threat_detection_enumeration
- cloudtrail_threat_detection_llm_jacking
- cloudtrail_threat_detection_privilege_escalation
- codeartifact_packages_external_public_publishing_disabled
- documentdb_cluster_public_snapshot
- ec2_ami_public
- ec2_ebs_default_encryption
- ec2_ebs_public_snapshot
- ec2_ebs_snapshot_account_block_public_access
- ec2_instance_account_imdsv2_enabled
- ec2_instance_port_cassandra_exposed_to_internet
- ec2_instance_port_cifs_exposed_to_internet
- ec2_instance_port_elasticsearch_kibana_exposed_to_internet
- ec2_instance_port_ftp_exposed_to_internet
- ec2_instance_port_kafka_exposed_to_internet
- ec2_instance_port_kerberos_exposed_to_internet
- ec2_instance_port_ldap_exposed_to_internet
- ec2_instance_port_memcached_exposed_to_internet
- ec2_instance_port_mongodb_exposed_to_internet
- ec2_instance_port_mysql_exposed_to_internet
- ec2_instance_port_oracle_exposed_to_internet
- ec2_instance_port_postgresql_exposed_to_internet
- ec2_instance_port_rdp_exposed_to_internet
- ec2_instance_port_redis_exposed_to_internet
- ec2_instance_port_sqlserver_exposed_to_internet
- ec2_instance_port_ssh_exposed_to_internet
- ec2_instance_port_telnet_exposed_to_internet
- ec2_securitygroup_allow_ingress_from_internet_to_high_risk_tcp_ports
- ecr_repositories_not_publicly_accessible
- glacier_vaults_policy_public_access
- guardduty_is_enabled
- iam_password_policy_expires_passwords_within_90_days_or_less
- iam_password_policy_lowercase
- iam_password_policy_minimum_length_14
- iam_password_policy_number
- iam_password_policy_reuse_24
- iam_password_policy_symbol
- iam_password_policy_uppercase
- kms_cmk_not_deleted_unintentionally
- kms_cmk_rotation_enabled
- neptune_cluster_public_snapshot
- opensearch_service_domains_not_publicly_accessible
- rds_instance_no_public_access
- rds_snapshots_public_access
- s3_account_level_public_access_blocks
- s3_bucket_policy_public_write_access
- s3_bucket_public_access
- s3_bucket_public_list_acl
- s3_bucket_public_write_acl
- securityhub_enabled
- sqs_queues_not_publicly_accessible

There are 55 available fixers.
```

1.4 PROWLER V5

: Prowler Studio를 필두로 한 AI 기능 확대 + MCP와 Fixer 기능 추가



Prowler Studio

Prowler Studio is an AI-powered toolkit for generating and managing security checks for Prowler. It is modular, model-agnostic, and supports multiple workflows and integrations.

> **Note:** The code generated by the AI system should be reviewed by a human before use.

Components Overview

Prowler Studio consists of several main components. Each is documented in detail in the `docs/` directory. Below is a summary of each component and a link to its full documentation.

1. Core (docs/core.md)

The foundational logic and workflows for check and fixer generation, check knowledge base using Retrieval Augmented Generation (RAG), and compliance mapping. Model-agnostic and designed for extensibility.

- **Main features:**
  - Modular workflow orchestration (LlamaIndex-based).
  - RAG dataset and semantic search in Prowler checks.
  - Provider abstraction for LLMs and embeddings.
- **Technical details and architecture:** docs/core.md

Prowler Studio

Prowler Lighthouse

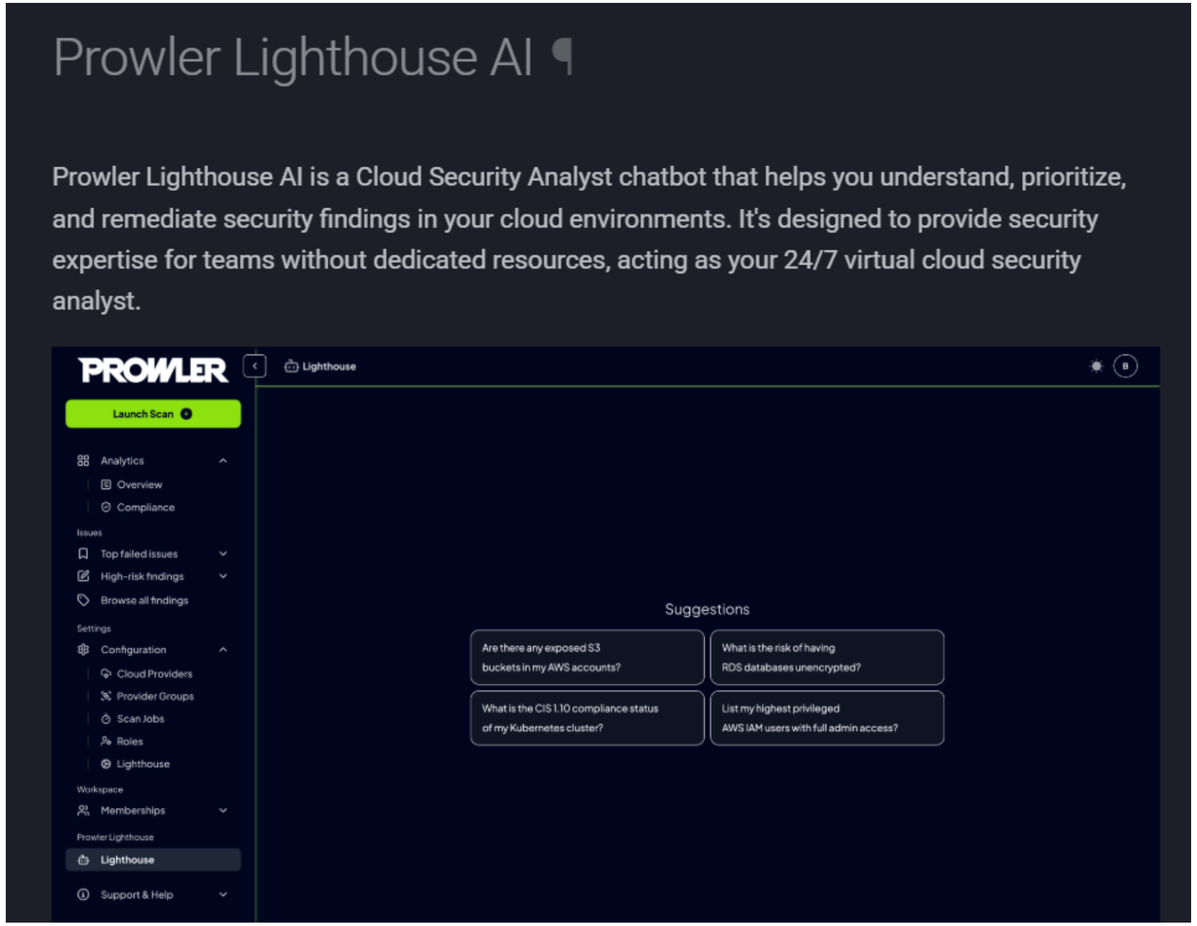
Prowler RAG를 이용한 AI 챗봇으로, 자연어 질의, 상세한 수정 가이드, 상황 맥락 기반 분석을 제공하는 멀티에이전트 기반 구조.

Prowler Cloud MCP

Google과 openAI API를 기반으로 한 MCP로, AI 코드 어시스트 통합(IDE, cursor 등)을 지원

1.4 PROWLER V5

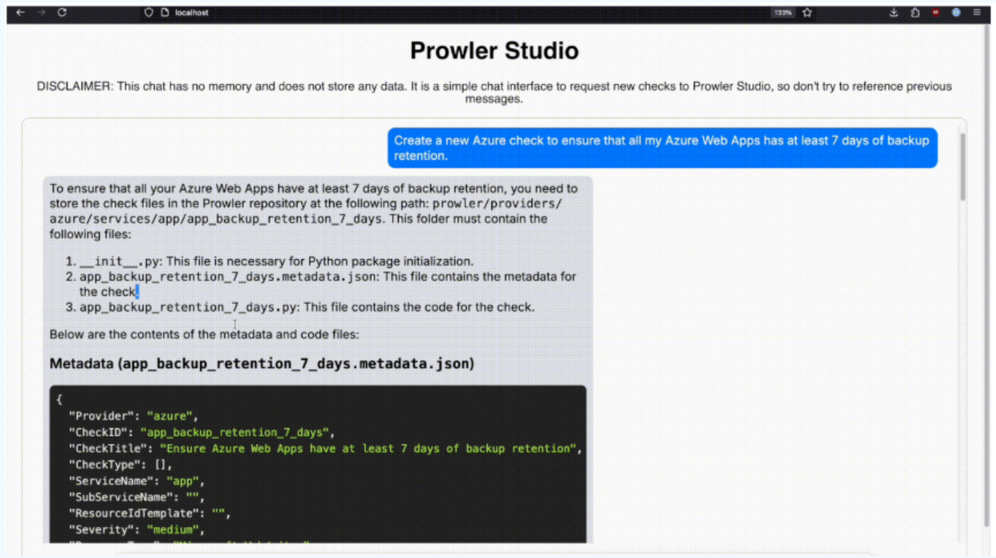
: Prowler Studio를 필두로 한 AI 기능 확대 + MCP와 Fixer 기능 추가



Prowler Lighthouse

- `prowler-studio` (includes Core + CLI by default)
- `prowler-studio-core`
- `prowler-studio-cli`
- `prowler-studio-api`
- `prowler-studio-mcp-server`

This release also introduces seamless integration with AI Code assists via MCP Server and comprehensive improved documentation for each component.



Prowler Cloud MCP

1.4 PROWLER V5

: LIGHTHOUSE를 필두로 한 MCP와 Fixer 기능 추가

Prowler Lighthouse / Prowler Cloud MCP

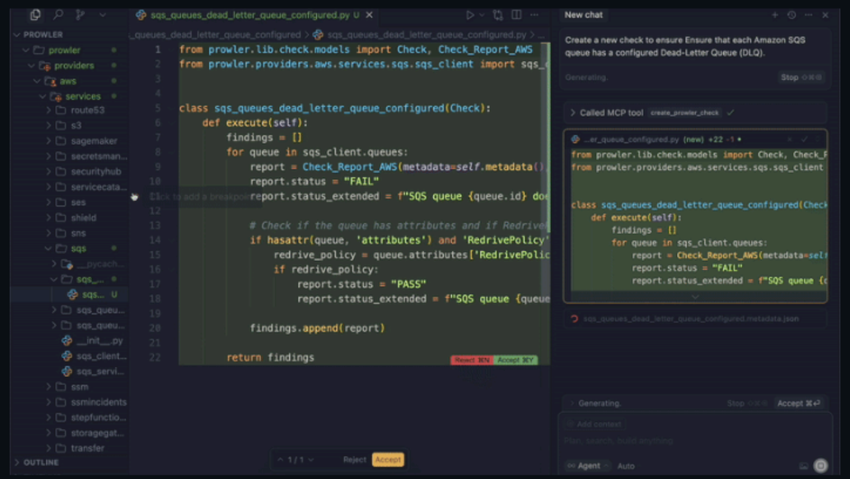
mcp_server chore: update core deps... 2 months ago

Prowler Studio MCP Server

Overview

The Prowler Studio MCP Server enables integration of Prowler Studio's AI-powered check and fixer generation into development environments via the Model Context Protocol (MCP). It is designed for seamless IDE integration (e.g., Cursor, VS Code).

Demo Time!



만들고 있던 기능을 공식 팀이 먼저 발표함!
어떡하지?

1.4 PROWLER V5

: LIGHTHOUSE를 필두로 한 MCP와 Fixer 기능 추가

Prowler Cloud MCP

Configuration

- Set `GOOGLE_API_KEY` (required) and `OPENAI_API_KEY` (optional) in your environment or `.env` file.
- Configure your IDE to connect to the MCP server (see below for examples).

Integration Examples

Cursor IDE

With Docker:

```
{
  "mcpServers": {
    "prowler-studio": {
      "command": "docker",
      "args": ["run", "--rm", "-e", "OPENAI_API_KEY=your_openai_api_key", "-e", "GOOGLE_API_KEY=your_google_api_key", "-i", "pr
    }
  }
}
```

1.4 PROWLER V5

: 공식 MCP가 있으면 우리는 어떻게 하지?

Prowler Cloud MCP

Configuration

- Set `GOOGLE_API_KEY` (required) and `OPENAI_API_KEY` (optional) in your environment or `.env` file.
- Configure your IDE to connect to the MCP server (see below for examples).

Integration Examples

Cursor IDE

With Docker:

```
{
  "mcpServers":
    "prowler-stu
      "command":
      "args": ["
    }
  }
}
```

Google API key/OpenAI API key가 필수적이고, IDE에 연결해서 쓰는 형식이네~
우리는 CLAUDE Desktop 용으로 만들고, IaC tool과 연동되게 만들자!

2. MCP 제작

2.1 기본 아이디어

2.2 Claude desktop MCP 제작

2.3 시연영상

2.1 기본 아이디어

: 컴플라이언스 점검 + 대응 + 알림이 한번에 되는 도구를 만들자!

컴플라이언스 점검 + 대응 + 알림이 한번에 되는 도구를 만들자!

점검

PROWLER

정보보호 및 개인정보보호 관리체계 인증
ISMS-P
Personal information & Information Security Management System

대응

Cloud Custodian

aws

알림

Security Hub

slack

2.1 기본 아이디어

: 컴플라이언스 점검 + 대응 + 알림이 한번에 되는 도구를 만들자!

왜 MCP여야 하는가?

Prowler는 CLI 중심(Cloud는 유료) → 비개발자 접근성이 낮음

로컬 데이터 파싱, Slack 알림, AWS Security Hub 연계 같은 기능을
하나의 **자연어 인터페이스**로 묶고 싶었음

현재 공식 MCP는 자체 개발한 UX를 사용함.
Claude Desktop을 이용하면 **시안성+확장성**
(다른 MCP를 같이 불러낼 수 있음) up

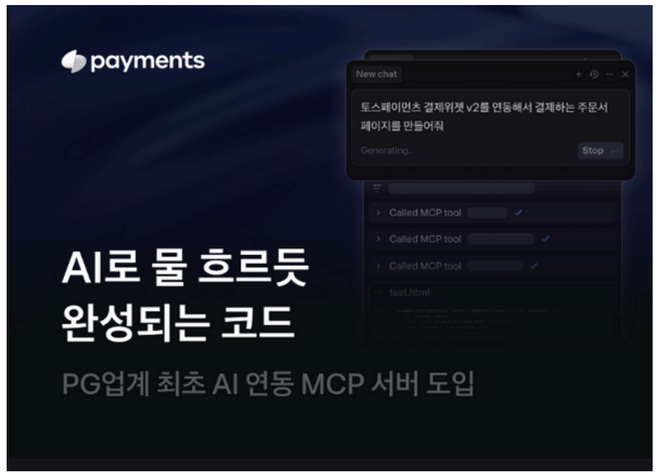
2.1 기본 아이디어

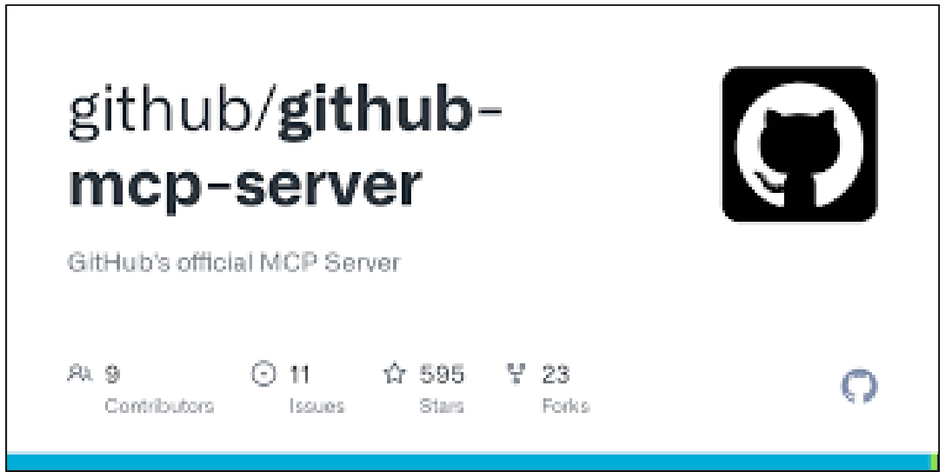
: 컴플라이언스 점검 + 대응 + 알림이 한번에 되는 도구를 만들자!

LLM에 우리 인스턴스 상태를 노출시켜도 안전한가?

당연히 완벽하게 안전하지는 않음 → **Local MCP제작 고려중**

그러나 일단은, prowler scan 결과에서 특별히 민감정보가 노출되지 않는다고 판단했음
+ 익명처리해서 해결해도 됨!





2.2 CLAUDE DESKTOP MCP 제작

점검

Prowler CLI로 scan해서 나온 로컬 파일

Claude

로컬 스캔 결과 불러오기, CSV·HTML·JSON 파일 파싱

컴플라이언스 결과를 자연어로 쉽게 풀어서 설명

점검 결과 내 Pass/Fail 항목을 사유와 함께 요약

Prowler CLI의 output 위치만 설정하면 — MCP가 알아서 파일을 읽고 사용자에게 설명/질의응답

2.2 CLAUDE DESKTOP MCP 제작

대응

Claude

- 컨텍스트와 함께 취약점 요약
- 가장 시급한 보안 문제 우선순위 지정
- 수정사항을 IaC 코드로 작성, YAML 파일 생성

Cloud Custodian

ANSIBLE

PROWLER FIXER

분석에서 실행까지 — 스캔 결과를 IaC 코드로 즉시 사용할 수 있는 수정 조치로 전환

2.2 CLAUDE DESKTOP MCP 제작

IaC 구현 툴 선택의 이유?

Custodian과 Ansible(개발중)은 운영 환경에서 변경 사항을 빠르게 반영하거나,
정책 기반의 자동 수정을 수행하는 데 적합하였음

특히 **Custodian**의 경우 **정책 기반** 관리 및 자동화가 가능해 최우선적으로 개발함

또한, Prowler fixer는 prowler에 최적화되어 있고, 기본제공된 스니펫을 사용할 수 있어
토큰 소모가 적었음(Claude의 개입 X)

2.2 CLAUDE DESKTOP MCP 제작

IaC 구현 툴 선택의 이유?

Cloud Custodian

클라우드 리소스를 **정책 기반으로** 자동 관리하는 오픈소스 툴,
개별 리소스에 대한 조건에 맞는 조치를 수행하는 데 최적화

정책 (Policy) 기반 관리

YAML 파일로 "어떤 리소스에, 어떤 조건일 때, 어떤 작업을 할지" 정의

우리 mcp는 이 YAML 파일을 만듦
ex) 퍼블릭 S3 버킷이 있으면 → 접근차단

```yaml
policies:
  - name: close-public-s3
    resource: aws.s3
    filters:
      - type: global-grants
        perms: [READ, WRITE]
    actions:
      - type: set-permissions
        block-public-acls: true
```

AWS, Azure, GCP, k8s+ 지원

aws

```yaml
policies:
  - name: my-aws-instances
    resource: aws.ec2
    filters:
      - type: value
        key: "tag:owner"
        value: "sam"
```

```python
import boto3

client = boto3.client('ec2')

custom_filter = [{
    'Name':'tag:owner',
    'Values': ['sam']}]

response =
client.describe_instances(Filters=custom_
filter)
```

동작 구조

1) 리소스 필터링
정책에서 조건 지정
(public: true, tag: Enviroment:dev)

2) 액션
조건에 맞는 리소스에 대해 조치 수행
(set-permission, delete, stop)

3) 로깅 & 보고
실행 결과를 json형태로 s3, CloudWatch 등에 저장

2.2 CLAUDE DESKTOP MCP 제작

왜 IaC 구현에서 Terraform은 안 했나요?

Terraform은 선언형 IaC로 인프라 전체를 관리하는 데 강점이 있지만,
단일 리소스·설정 변경을 빠르게 반영하는 시나리오에서는
Custodian/Ansible이 더 단순하고 효율적.

MCP에서 자연어 → 액션 변환 시, Custodian/Ansible은
단일 Playbook/Policy 파일 생성으로 즉시 실행 가능.

이번 버전은 보안 점검 → 즉시 대응이 목적이어서
필요한 리소스만 신속히 수정 가능한 툴을 선택.

2.2 CLAUDE DESKTOP MCP 제작

알림

Cloud Custodian

ANSIBLE

PROWLER FIXER

slack

MCP 도구의 파일 읽기/쓰기 작업 알림

생성한 YAML 파일을
dry-run/run 후 알림

YAML 파일의 주요 내용 알림

Security Hub

IaC 도구, Prowler, MCP가 제작한
파일들을 오케스트레이션

모든 MCP 플로우와 IaC 드라이런 결과를 Slack 알림으로 전달

2.2 CLAUDE DESKTOP MCP 제작

MCP에서 dry run 자동 기능을 제거하고 별도 대시보드를 제작한 이유?

처음에는 “딸깍” 하면 모든 대응이 자동화되는 도구를 만들고자
대화 안에서 Custodian Run까지 되게 하려고 하였음

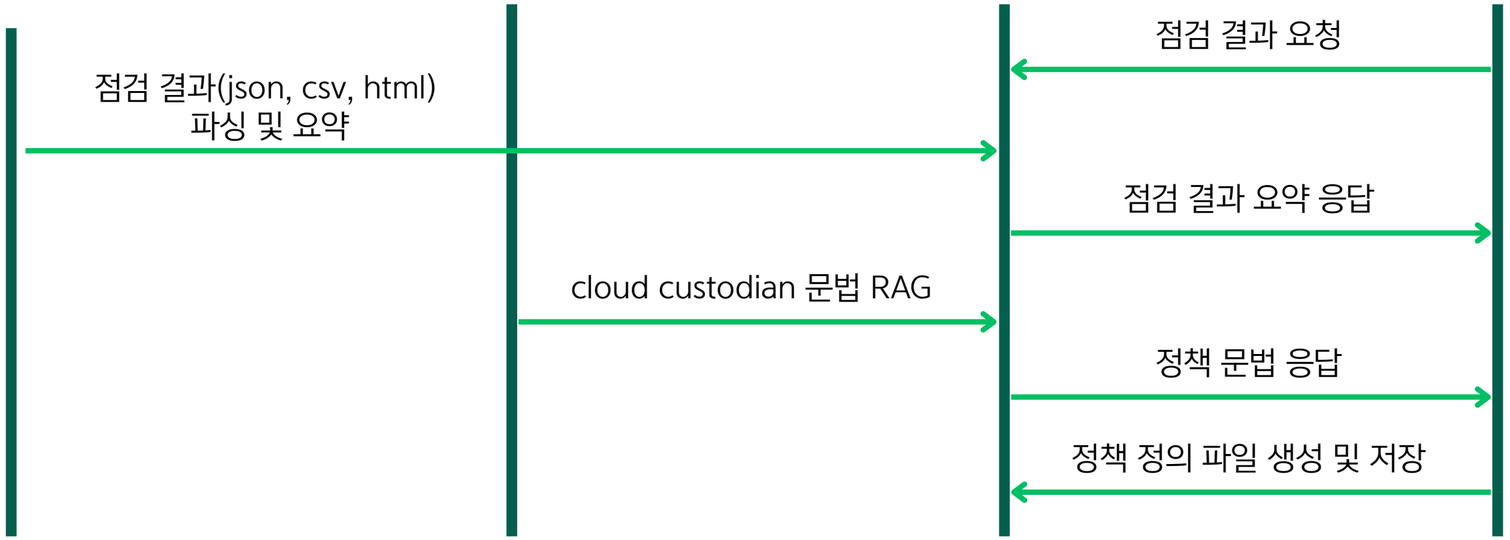
보안적으로 안전하지 않다고 팀 내부에서 판단 → MCP는 custodian yaml 파일까지 제작,

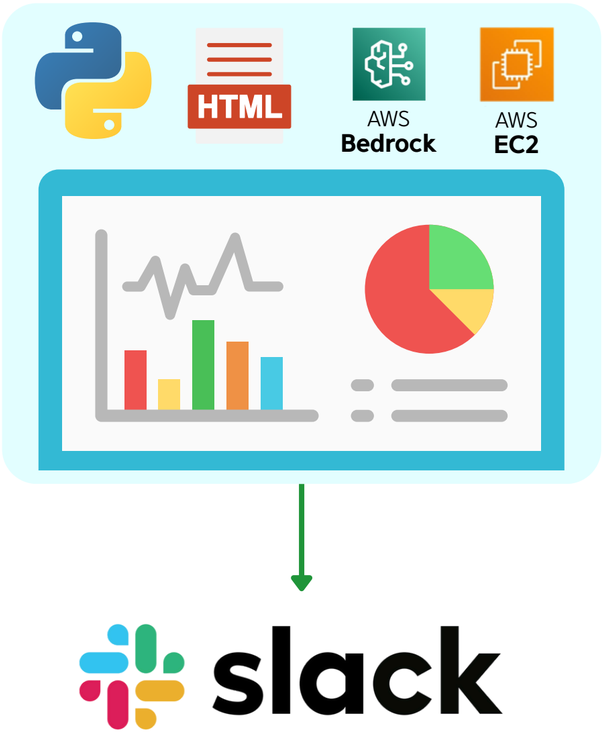
yaml 파일에 사용자가 credentials을 직접 추가한 후 대시보드에서 dry run할 수 있게 함

파일 읽기/쓰기 + yaml 생성 등 중요한 워크플로우 및 주요 위험사항 등 중요 보안내용은
Slack 알림까지!

2.2 CLAUDE DESKTOP MCP 제작

prowler Cloud Custodian FastMCP





2.2 CLAUDE DESKTOP MCP 제작

: AI 및 SASS 사용의 허가

제작한 MCP의 tool(함수) :

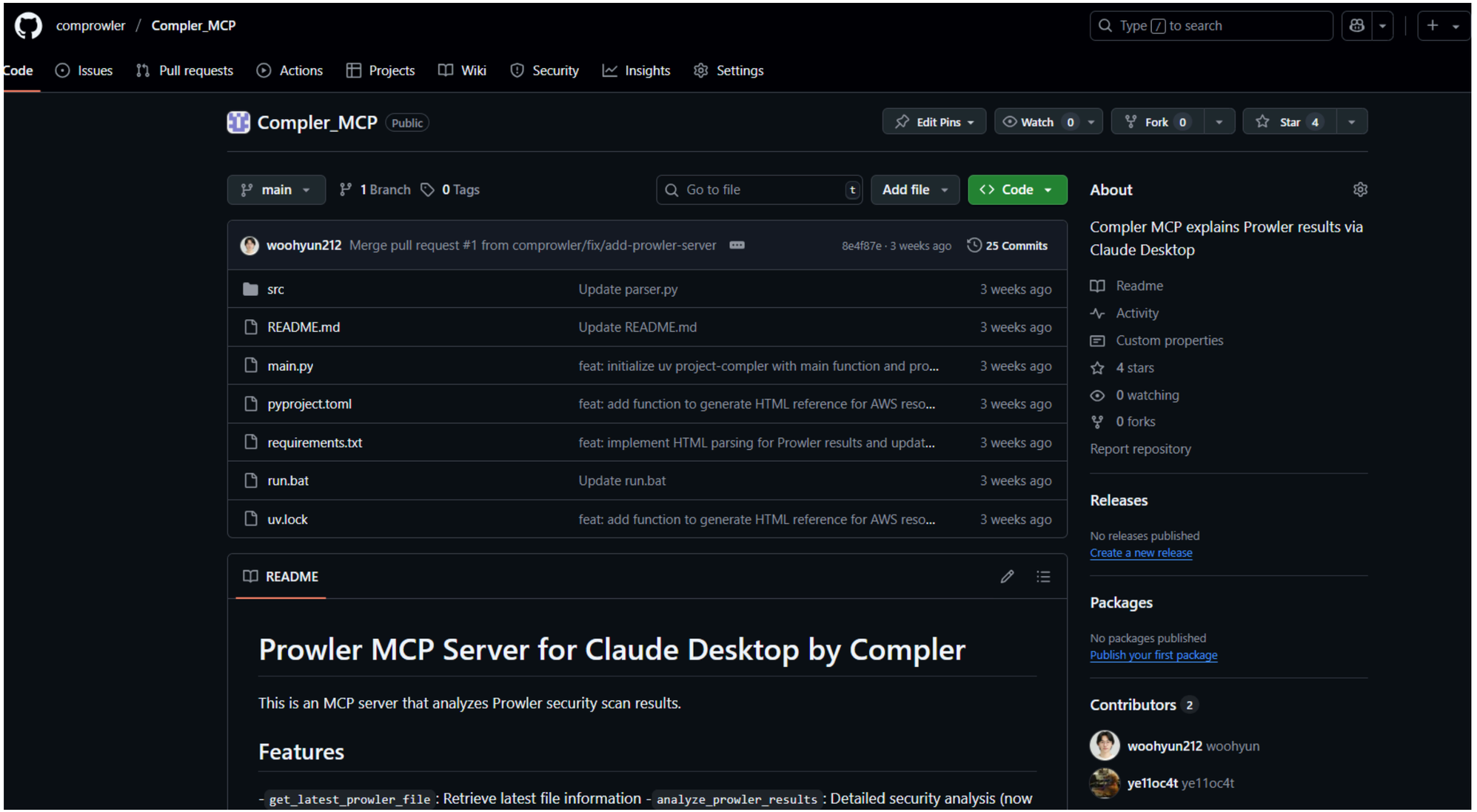
- get_latest_file
- analyze_html_file
- analyze_csv_file
- analyze_json_file
- analyze_prowler_results
- get_prowler_reports_list

- get_security_summary
- get_cloud_custodian_aws_resource_reference_html
- write_yaml_file
- create_yaml_directory
- list_iac_files
- get_iac_file_content

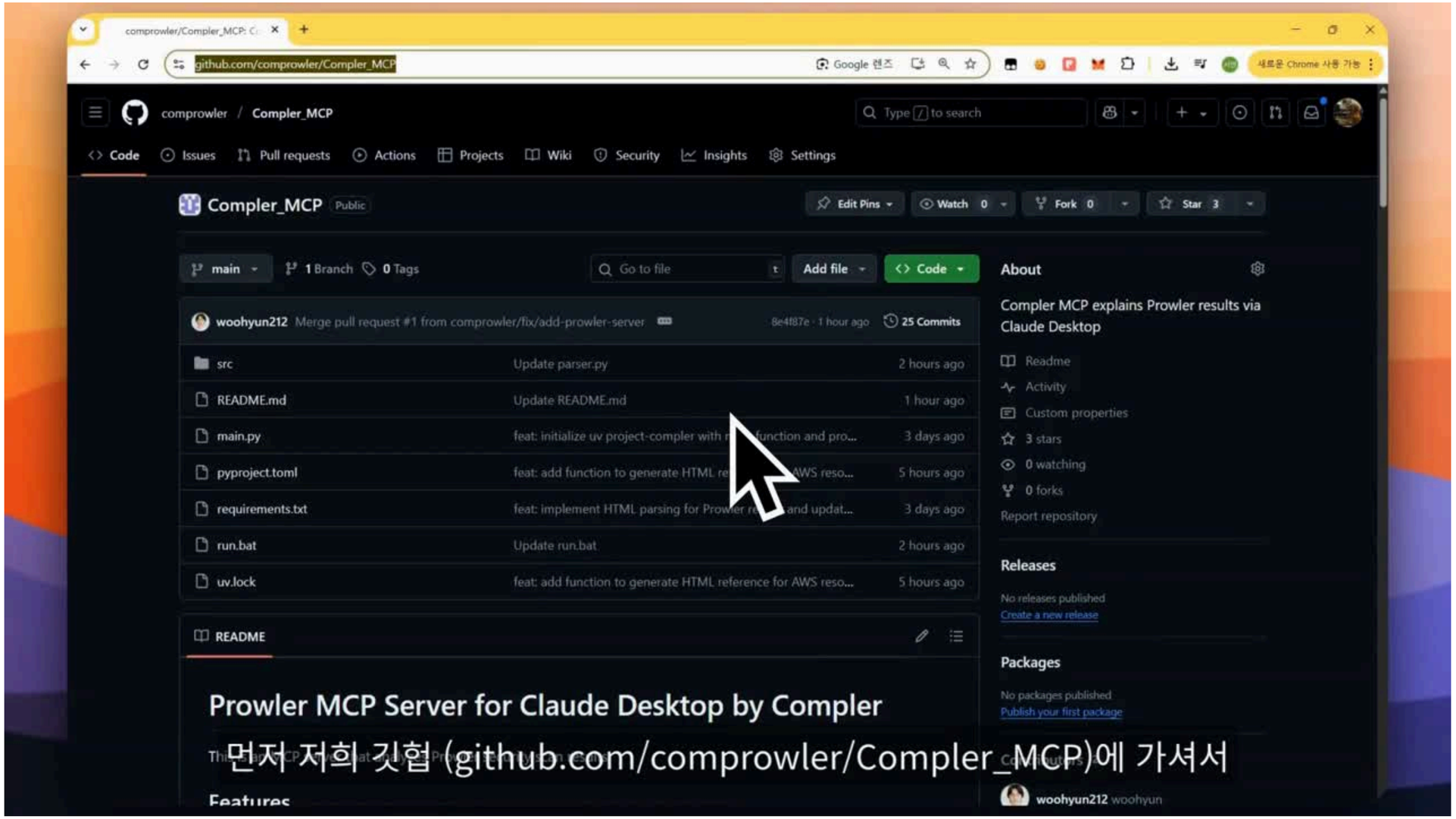
notify slack,
auto dry run IaC,
analyze IaC run results

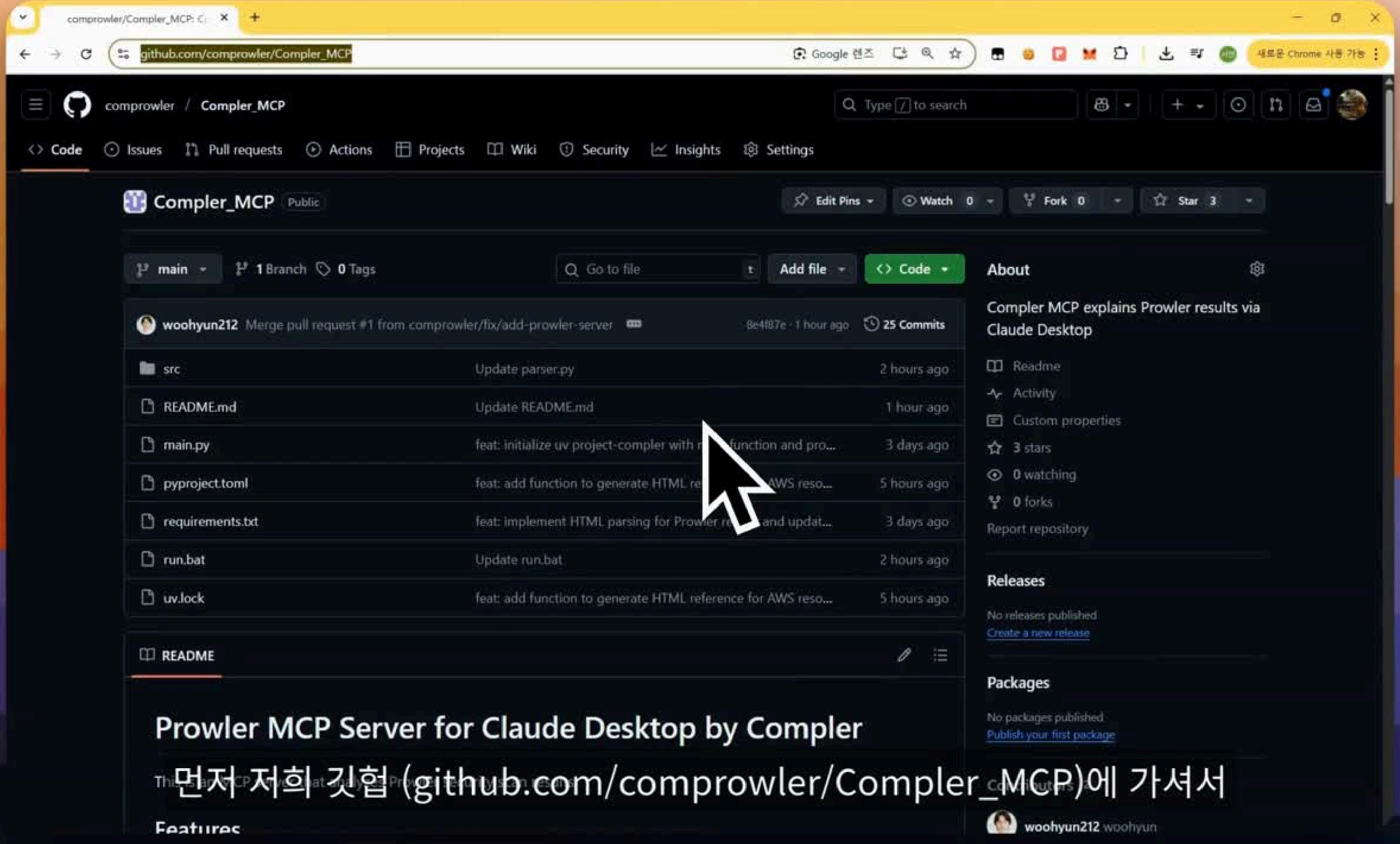
2.2 CLAUDE DESKTOP MCP 제작

: 오픈소스로 공개!



2.4 시연영상


comprowler/Compler_MCP
github.com/comprowler/Compler_MCP
comprowler / Compler_MCP
Code Issues Pull requests Actions Projects Wiki Security Insights Settings
Compler_MCP Public
Edit Pins Watch 0 Fork 0 Star 3
main 1 Branch 0 Tags Go to file Add file Code
woohyun212 Merge pull request #1 from comprowler/fix/add-prowler-server 8e4f87e · 1 hour ago 25 Commits
src Update parser.py 2 hours ago
README.md Update README.md 1 hour ago
main.py feat: initialize uv project-compler with 3 days ago
pyproject.toml feat: add function to generate HTML 5 hours ago
requirements.txt feat: implement HTML parsing for Prowler 3 days ago
run.bat Update run.bat 2 hours ago
uv.lock feat: add function to generate HTML reference for AWS reso... 5 hours ago
README
Prowler MCP Server for Claude Desktop by Compler
Features
About
Compler MCP explains Prowler results via Claude Desktop
Readme
Activity
Custom properties
3 stars
0 watching
0 forks
Report repository
Releases
No releases published
Create a new release
Packages
No packages published
Publish your first package
먼저 저희 깃헙(github.com/comprowler/Compler_MCP)에 가셔서



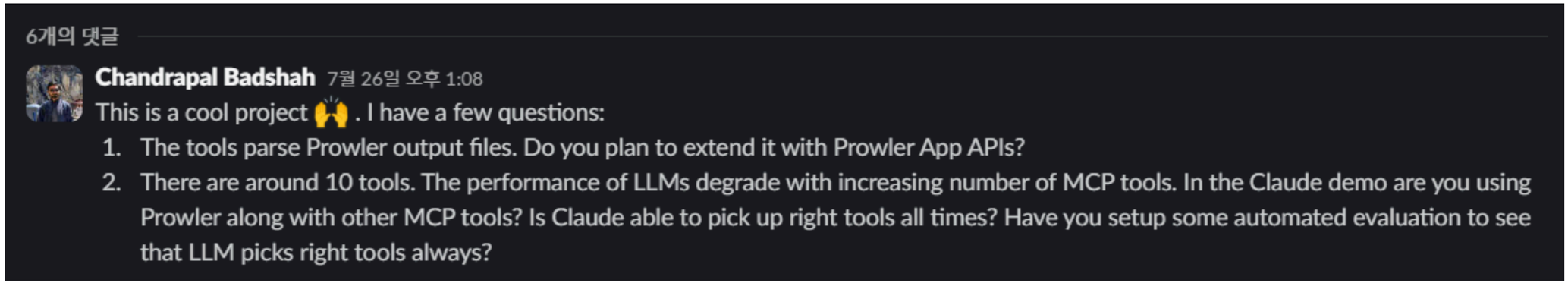
먼저 저희 깃헙 (github.com/comprowler/Compler_MCP)에 가셔서

3. CLOSING

3.1 향후계획

3.2 향후계획

:커뮤니티에서 요청받은 기능 추가 및 QA 진행, 가능하다면 망분리 환경에서 사용할 수 있는 로컬 MCP까지


6개의 댓글

Chandrapal Badshah 7월 26일 오후 1:08

This is a cool project 🙌 . I have a few questions:

1. The tools parse Prowler output files. Do you plan to extend it with Prowler App APIs?
2. There are around 10 tools. The performance of LLMs degrade with increasing number of MCP tools. In the Claude demo are you using Prowler along with other MCP tools? Is Claude able to pick up right tools all times? Have you setup some automated evaluation to see that LLM picks right tools always?

Prowler App API 지원 확대

MCP tool 통합

비슷한 기능을 가진 MCP 도구들을 병합해 claude에 노출되는 도구의 개수를 줄여 간소화 및 최적화

사용자 가이드 제공

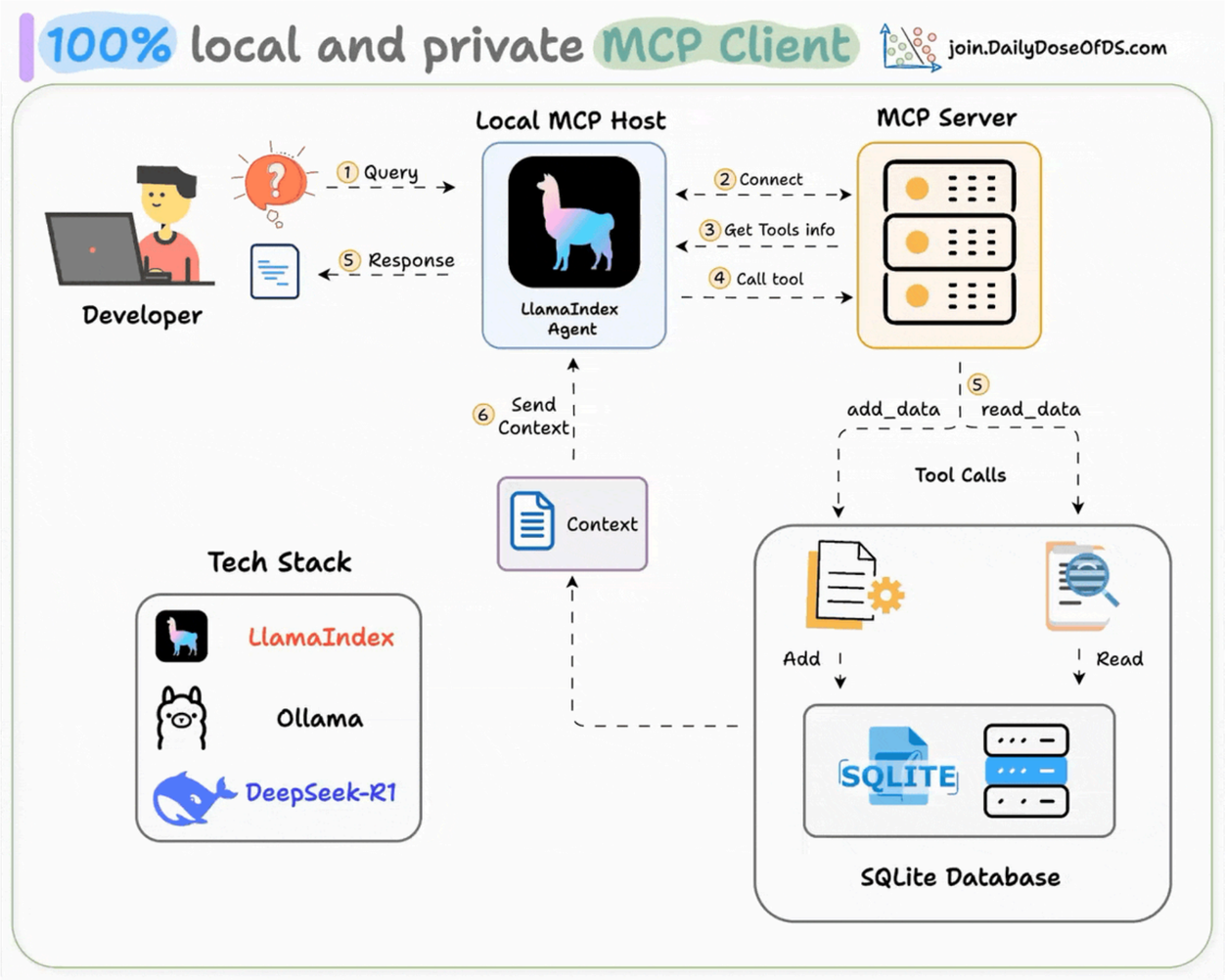
최적의 결과를 얻을 수 있도록
프롬프트 가이드라인과
효과적인 프롬프트 예시 제공 예정

품질 보증(QA) 테스트 강화

도구가 올바르게 호출되는지 확인
MCP 도구 선택 문제 및
Claude의 환각(hallucination) 가능성 최소화

3.2 향후계획

:커뮤니티에서 요청받은 기능 추가 및 QA 진행, 가능하다면 망분리 환경에서 사용할 수 있는 로컬 MCP까지



#AWSKRUG

Prowler MCP 개발 및
AWS 취약점 점검 / 조치 자동화 아키텍처
구성기

Thank you for Watching!

#AWSKRUG

Prowler MCP 개발 및 AWS 취약점 점검 / 조치 자동화 아키텍처 구성기

Q&A