Verifiche di compliance in ambienti Cloud

Presentazione dell'elaborato finale

Laurea Triennale in Sicurezza dei Sistemi e delle Reti Informatiche

Niccolò Volontè (20642A)

June 27, 2025

UNIVERSITÀ DEGLI STUDI DI MILANO

Introduzione e contesto

Cloud Computing e Compliance

- Evoluzione del Cloud Computing
- Distinzione *IaaS*, *PaaS* e *SaaS*
- Amazon Web Services (*AWS*)
- Compliance come sfida

Obiettivi del lavoro

Suite di sonde

- Sviluppo di sonde che verificano la compliance
- Integrazione con la piattaforma MoonCloud
- Adeguamento a standard di sicurezza
- Automatizzazione della verifica

La compliance nel cloud

Significato e importanza

- Definizione di compliance
- Rilevanza per la sicurezza informatica
- Sfide specifiche del cloud

Standard di riferimento

Enti, direttive e normative

- Center for Internet Security (*CIS*) - CIS AWS Foundations Benchmark
- National Institute of Standards and Technology (*NIST*) - NIST SP 800-53

Tecnologie utilizzate

Linguaggi e strumenti

- *Python* come linguaggio di programmazione
- `Boto3` per l'interazione con AWS
- *MoonCloud* come piattaforma di integrazione

Esempio di utilizzo di Boto3

```
import boto3
client = boto3.client(
 'sqs',
 region_name='eu-central-1',
 aws_access_key_id='YOUR_ACCESS_KEY',
 aws_secret_access_key='YOUR_SECRET_KEY'
)
response = client.list_queues()
```

MoonCloud

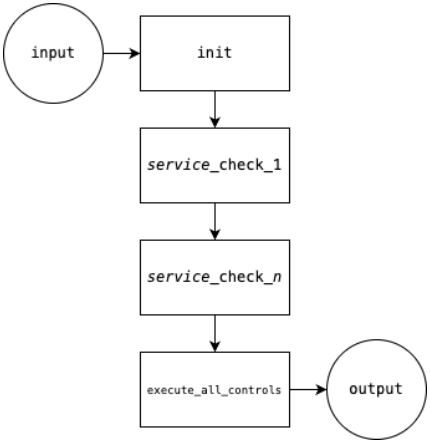
Piattaforma, funzionalità e architettura

- Esegue sonde di assurance su infrastrutture ICT
- Architettura basata su immagini Docker e CI/CD
- Dashboard per la gestione dei target, credenziali e risultati
- Modello a stati finiti: *forward*, *rollback*

Struttura di una sonda

Componenti per la creazione

- Codice Python all'interno del file `probe.py`
- `schema.json` e `test.json` per input validato
- `Dockerfile`, `.gitlab-ci.yml` per la pipeline
- Struttura con `atoms` eseguiti in sequenza
- Output *strutturato*



Esempio di sonda

Sonda `aws_sqs`

- Controlli relativi a crittografia, tagging, policy pubbliche
- Basata su controlli di AWS Security Hub
- Esempio di scansione multiregione
- Integra parametri personalizzati via dashboard

Implementazione multiregione

```
self.clients = {}
for idx, region in
enumerate(specific_regions, start=1):
 self.clients[f'client_{idx}'] =
boto3.client(
 'sqs',
 aws_access_key_id='YOUR_ACCESS_KEY',
 aws_secret_access_key='YOUR_SECRET_KEY'
 region_name='eu-central-1',
)
```

aws_vulnerability

Sonda per la gestione CVE

- Sonda custom che elenca CVE trovate da AWS Inspector
- Analisi di Elastic Container Registry (*ECR*), Elastic Compute Cloud (*EC2*) e *Lambda* functions
- Supporta una visione dinamica del rischio

Esecuzione e integrazione

Gestione I/O e deploy

- Input via JSON schema con validazione
- Gestione sicura delle credenziali
- Pipeline CI/CD su GitLab per ogni sonda
- Integrazione tramite backend di MoonCloud

Risultati ottenuti

Visualizzazione sulla dashboard

- Controlli strutturati in blocchi
- Risultato numerico e descrittivo
- Sommario con percentuale di conformità
- Log dettagliato con eccezioni gestite

Competenze acquisite

Strumenti, metodologie e conoscenze

- Tecnologie: Python, AWS, Docker, GitLab CI/CD
- Sviluppo modulare e orientato alla sicurezza
- Analisi di documentazione tecnica
- Esperienza in un progetto reale su una piattaforma in uso

Conclusione e sviluppi futuri

Riflessioni e prospettive

- Sviluppo di 13 sonde operative
- Contributo concreto all'evoluzione di MoonCloud

- Implementazione multiregione
- Estensione a nuovi benchmark e cloud provider

Verifiche di compliance in ambienti Cloud

Grazie per l'attenzione!