DevSecOps

The key of continuous security

2019@DevOpsDays

Whois

I am Anderson

MaiCoin Security Engineer
AWS Certified All-5 + Sec specialty
Too many interests...

4ndersonLin

Agenda

- Sec with DevOps
- About continuous security
- Monitor and Automation
- Conclusions

DevOps + Sec = ?

Make DevOps secure

DevOps 帶給我們自動化的機會

Sec 帶給我們**安全**的機會

Make DevOps secure ~~or die~~

哪些人需要同時擁抱DevOps和Sec。

Make DevOps secure ~~or die~~

於是，這些~~怕死的~~人就試著把它們合在一起。

DevSecOps

Why DevSecOps?

軟性需求

追求流程安全性

追求產品安全性

減少Sec和DevOps的隔閡

剛性需求(~~沒有會死~~)

合規性

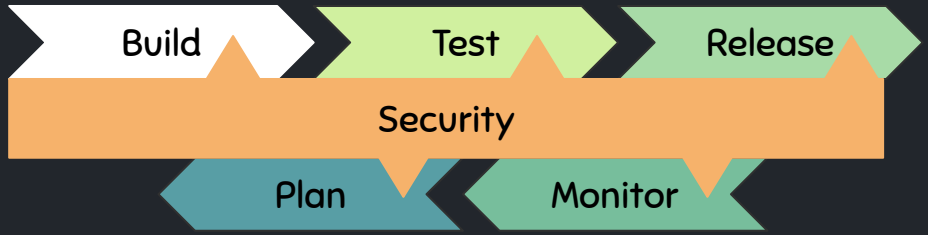
資產價值高

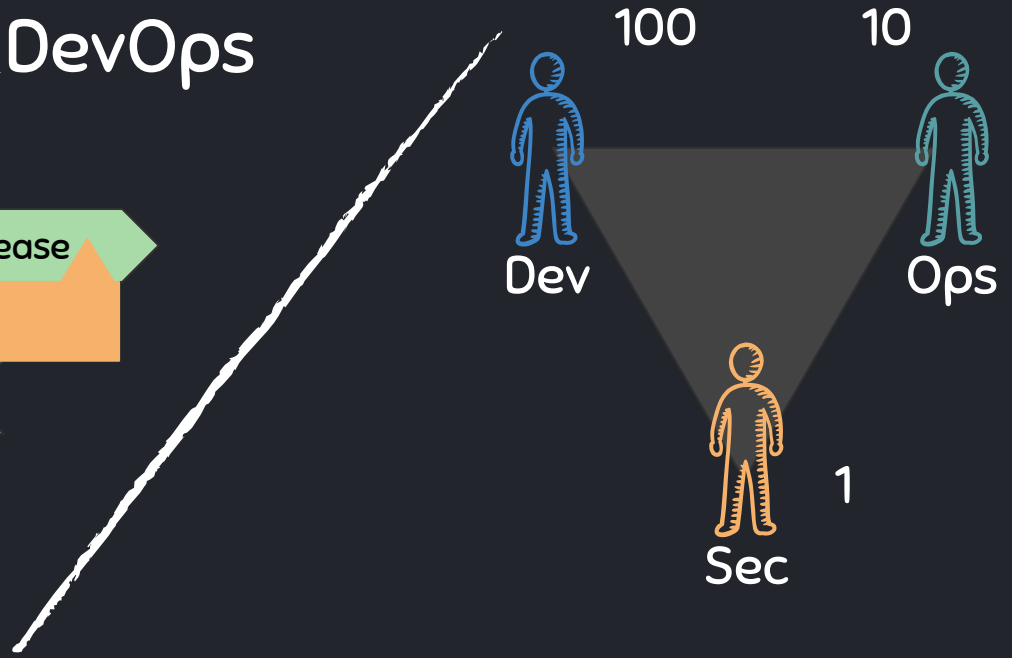
資安風險高

Why DevSecOps?

Two faces of DevSecOps

1. 確保DevOps技術和方法的安全性
2. 資安人員融入和適應DevOps





Chance of DevSecOps

Build/Test faster

所以... 我們有機會更早/快發現漏洞

Release/deploy faster

所以... 我們有機會更快的修補漏洞

More monitoring details

所以... 我們有機會更快的反應問題

Challenge of DevSecOps

傳統的資安工具不支援...

- Cloud service : Native solution
- Container : New tools

難以保護的pipeline...

- Protect the credentials first

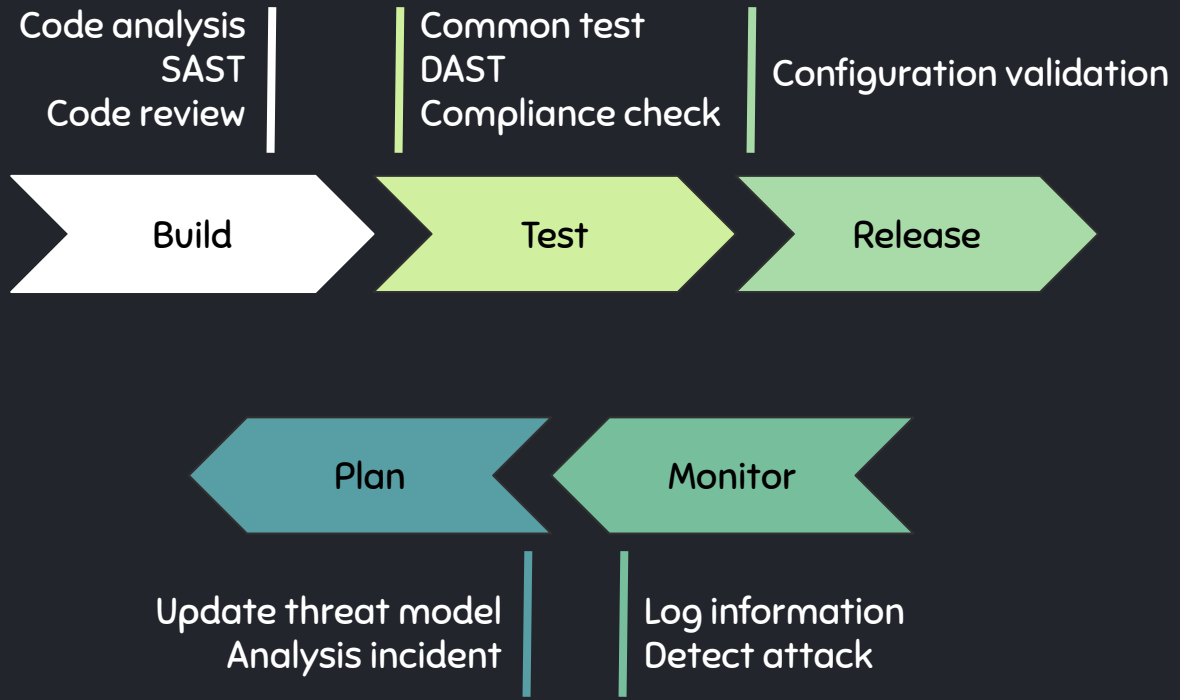
資安人擁抱自動化從何開始...

- 熟悉的工具開始 : SAST, DAST
- 自動化監控和回應

Well Known

DevSecOps life cycle

LIFECYCLE


Code analysis
SAST
Code review
Common test
DAST
Compliance check
Configuration validation
Build
Test
Release
Plan
Monitor
Update threat model
Analysis incident
Log information
Detect attack

Under the Application

在Application之下，關於Infrastrcture的部分，

我們有沒有機會用類似的方法，持續的保持它的安全性呢?

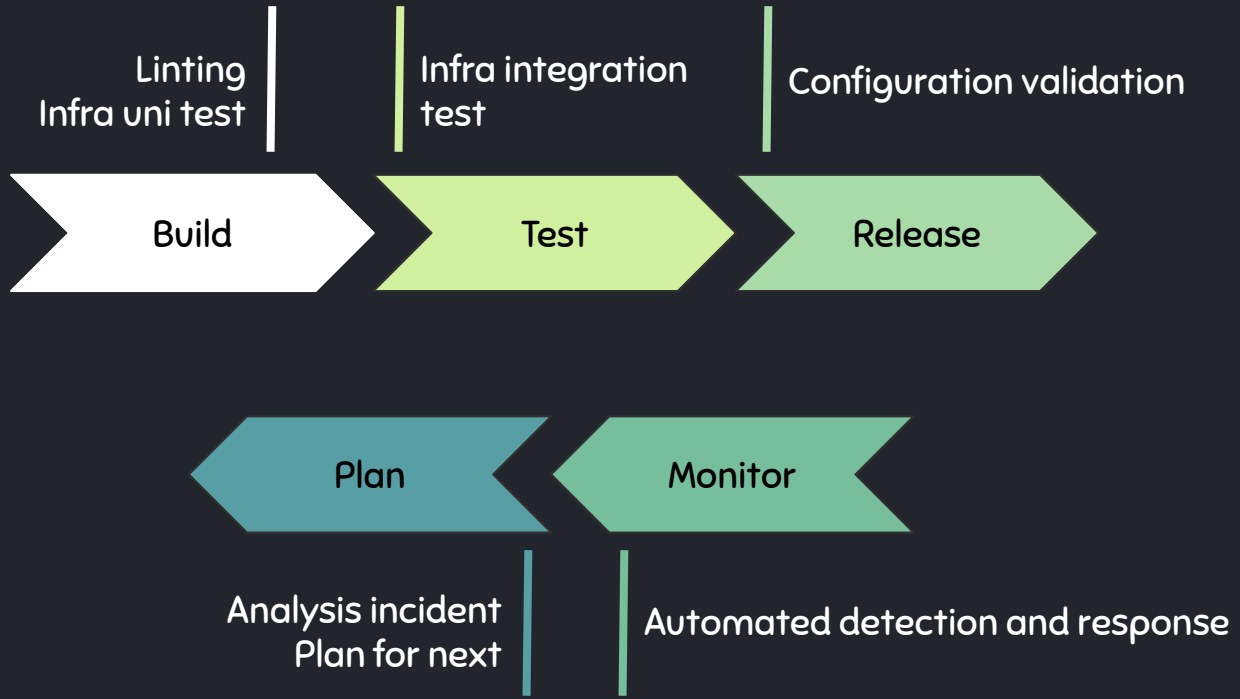
Yes, we can.

Before we run

我們需要準備什麼?

1. Everything as Code(IaC, PaC)
2. Secure by default
3. 請使用有提供API的工具和服務
4. 融入追求安全的精神

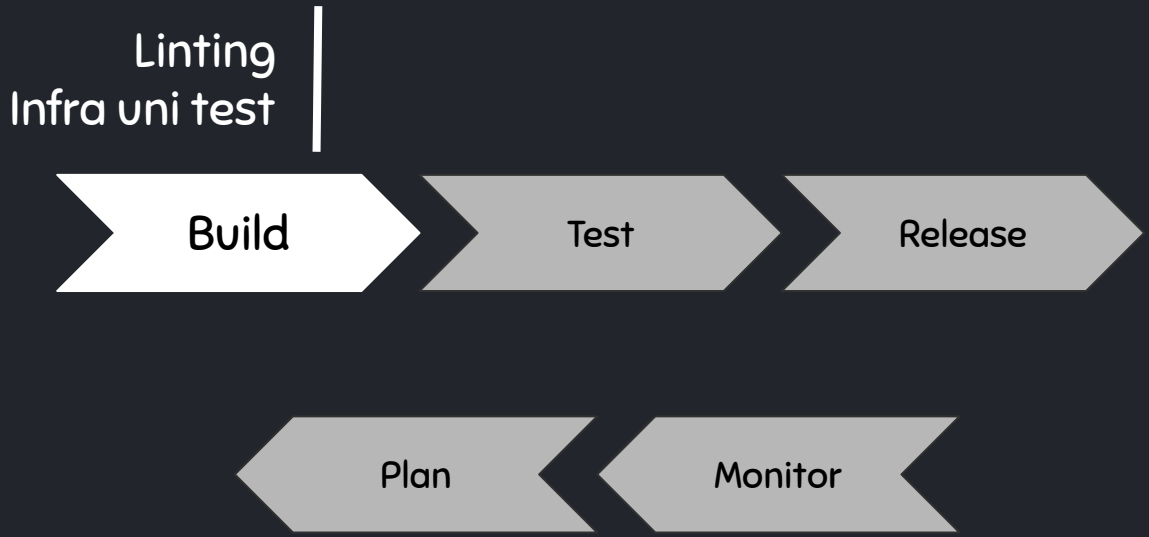
Continuous security



Continuous Security

Build phase

Life cycle : Build phase



Pro and Con

Pros:

- 可以測試到每個resource
- 確保產出的Code具備最基本的可靠性

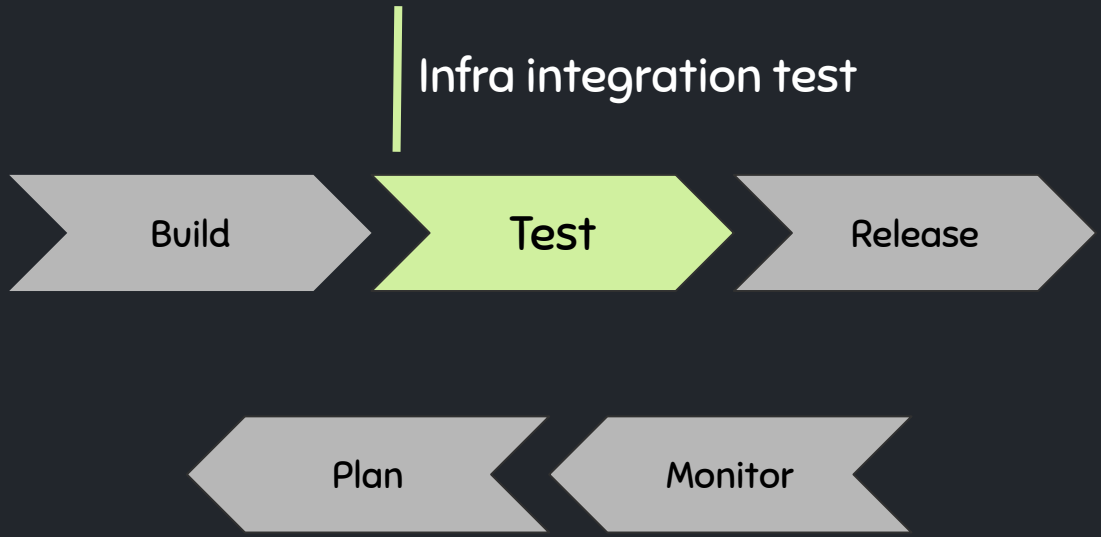
Cons:

- 逐一寫出的測試後續難以維運
- 覆蓋面可能不夠廣泛

Continuous Security

Test phase

Life cycle : Test phase



Life of Test phase

Infra as Code > Testing

- Terratest
- Cloudformation validation pipeline

Pro and Con

Pro:

- 有較完整的測試，和預期結果的偏差較小。

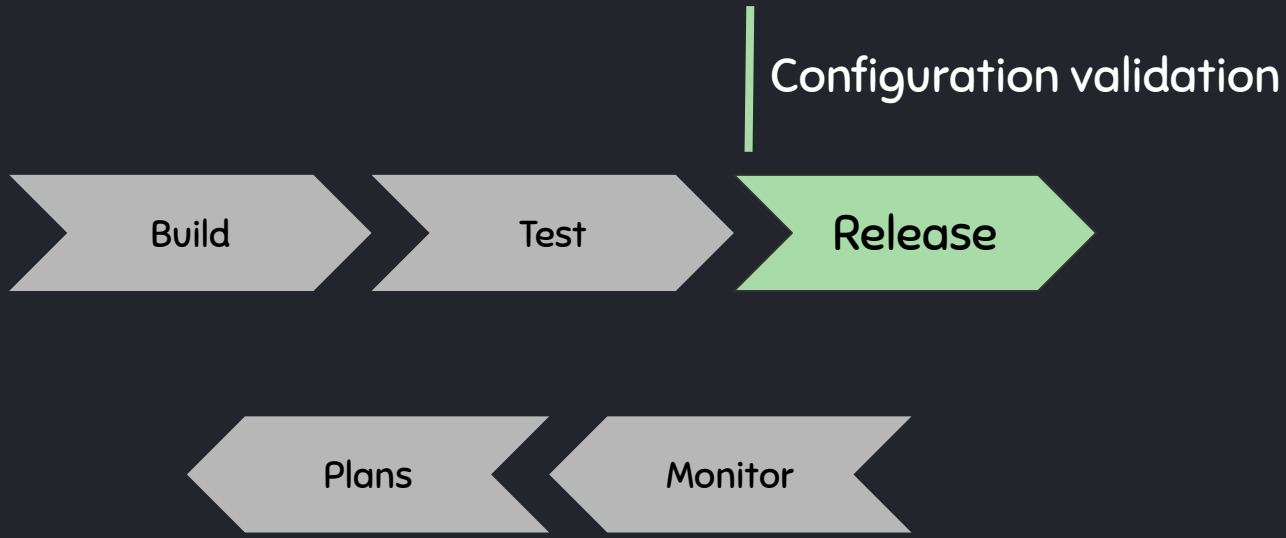
Con:

- 還不足以確認完全符合Compliance和Policy的需求

Continuous Security

Release phase

Life cycle : Release phase



Life of Release phase

Policy as code

- AWS Config
- Chef InSpec
- Open policy agent

Pro and Con

Pro:

- 確保了產出的系統設定的符合預定的policy

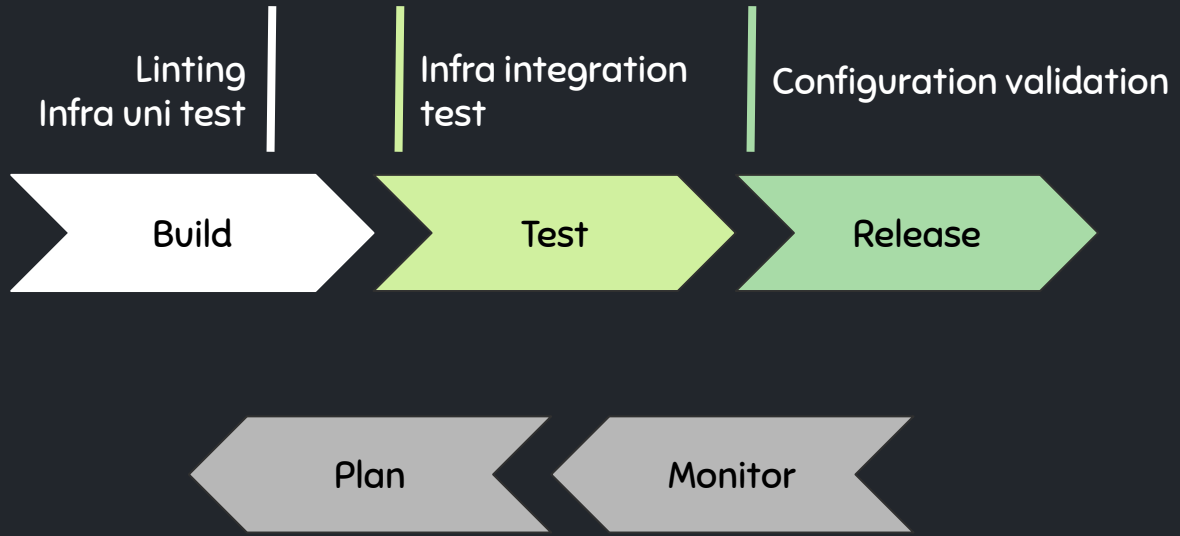
Con:

- 每個平台和工具有不同的DSL要學

Take a break

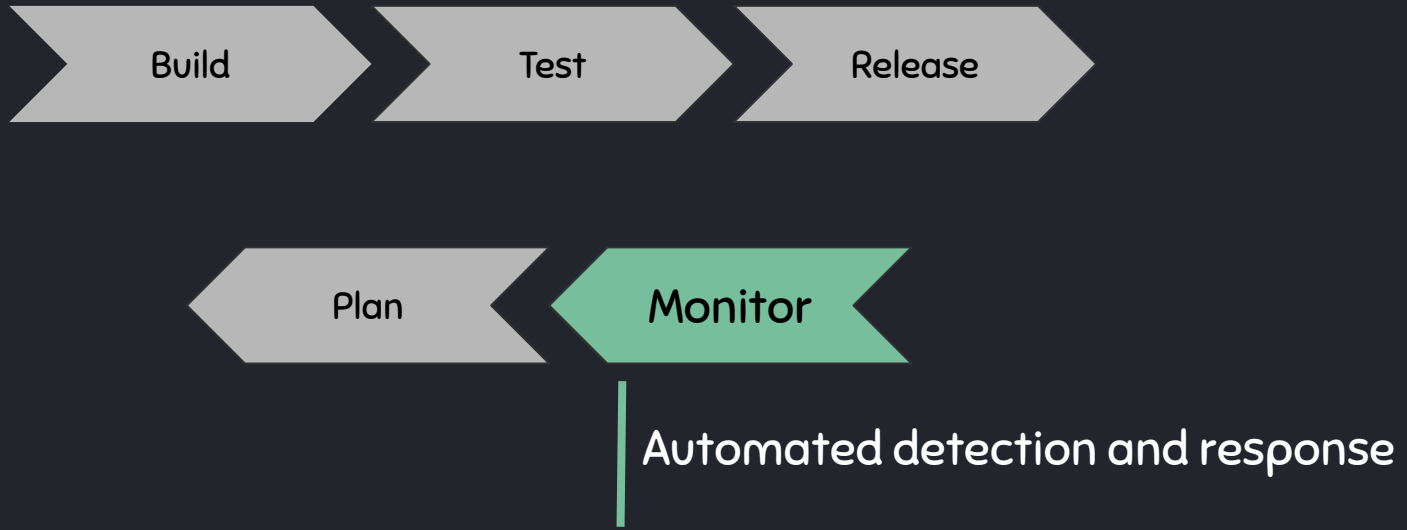
Recap

Recap of life cycle



Monitor and Automation

Life cycle : Monitor phase



Why we need to do this

確保持續的一致性和合規性

對抗風險：

- 未授權的行為
- 惡意攻擊
- 違反Compliance rule

...

Monitor and Automation

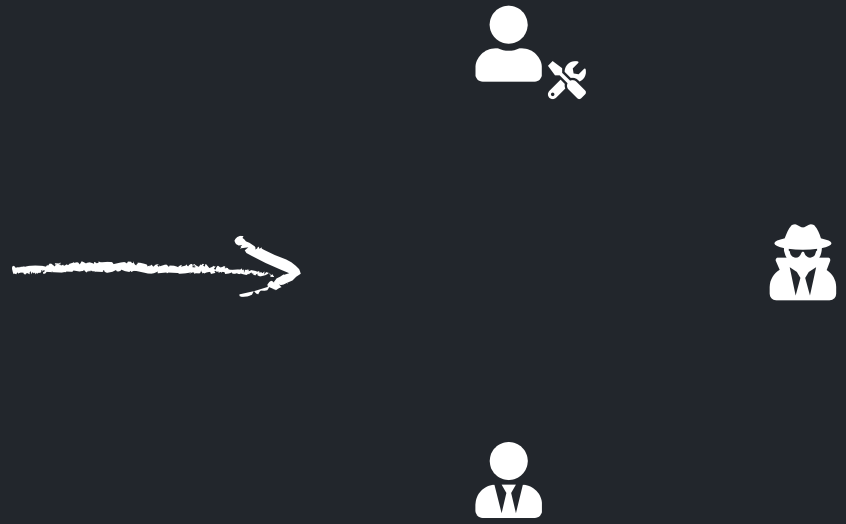
自動化偵測並採取行動

Detection

偵測的內容:

- 異常變更
- 異常存取
- 惡意攻擊
- PCI requirement
- CIS control

...

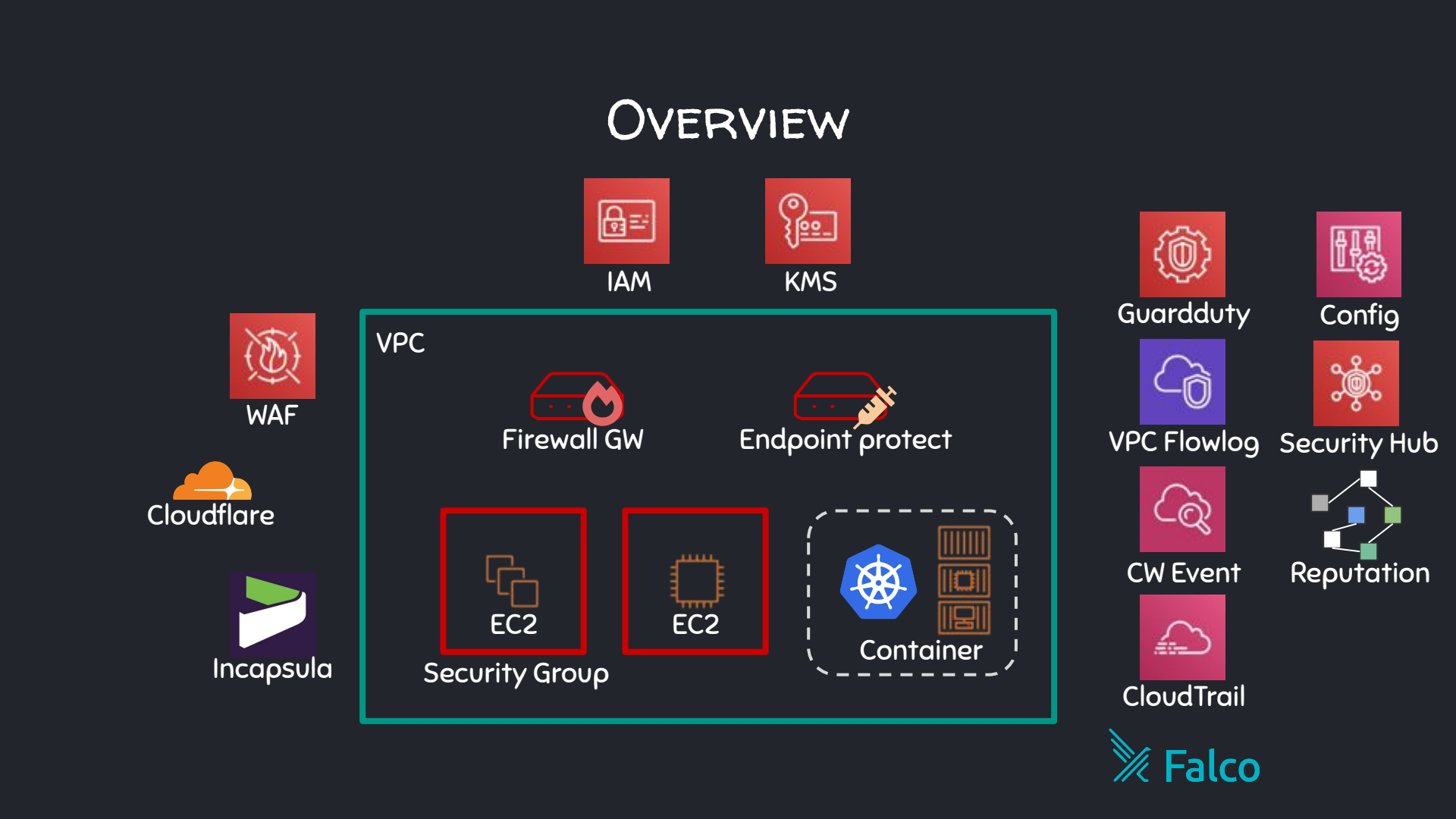


Action

依據嚴重程度不同，自動採取不同行為:

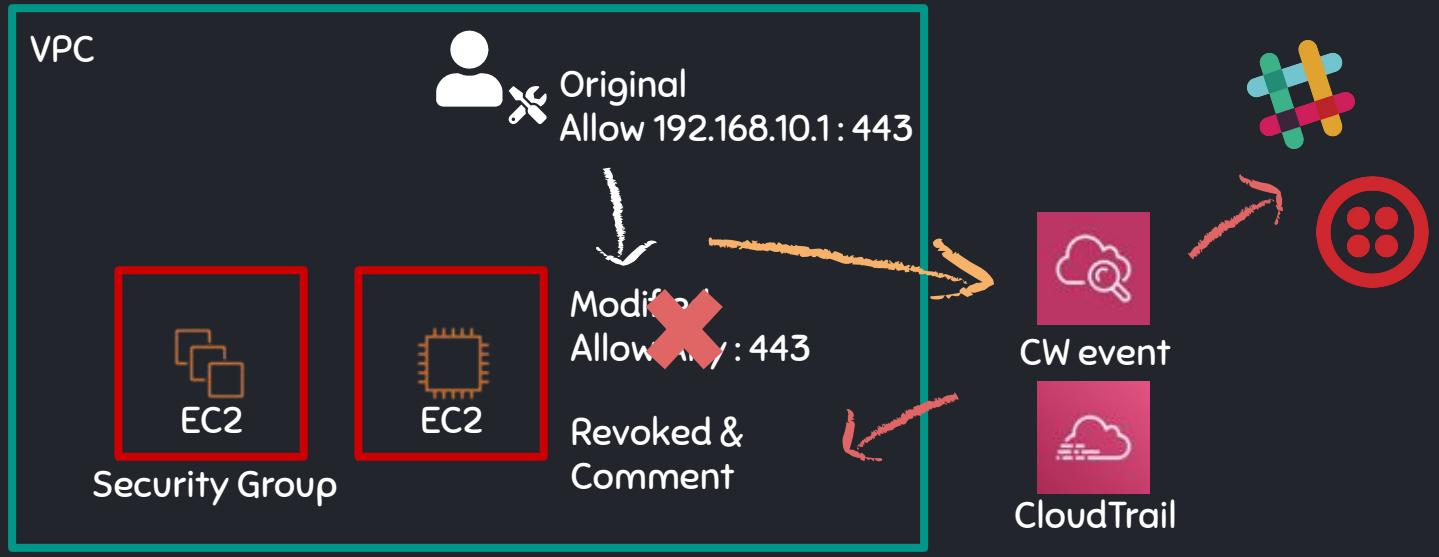
- 告警
- 還原變更
- 關閉/隔離/移除異常個體
- 停用/刪除User或API Key
- 建立Reputation list(整合內部和外部的情資)

Overview


IAM
KMS
Guardduty
Config
WAF
VPC
VPC Flowlog
Security Hub
Firewall GW
Endpoint protect
Cloudflare
CW Event
Reputation
EC2
EC2
Container
Incapsula
Security Group
CloudTrail
Falco

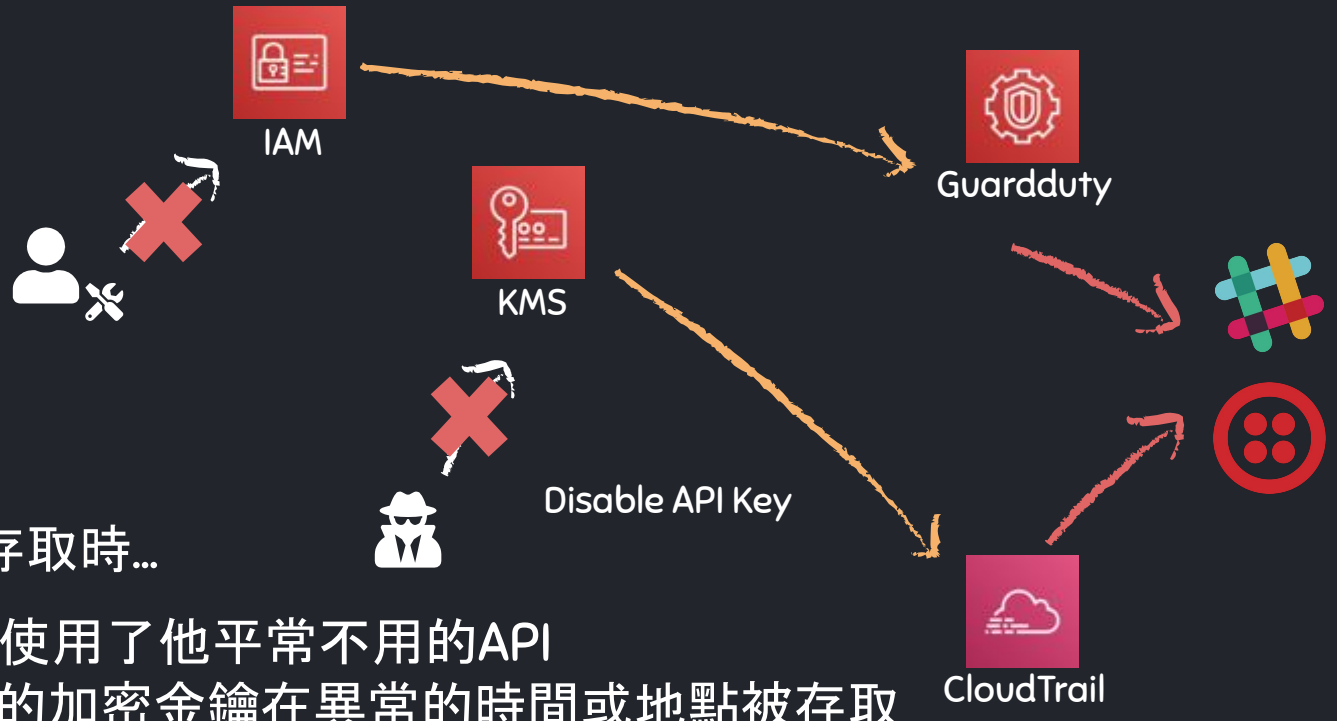
異常變更

發生可能由維運人員手動或是未經審核的變更時...
例如Security Group規則突然被異動



https://github.com/4ndersonLin/serverless-sg-sentry

異常存取

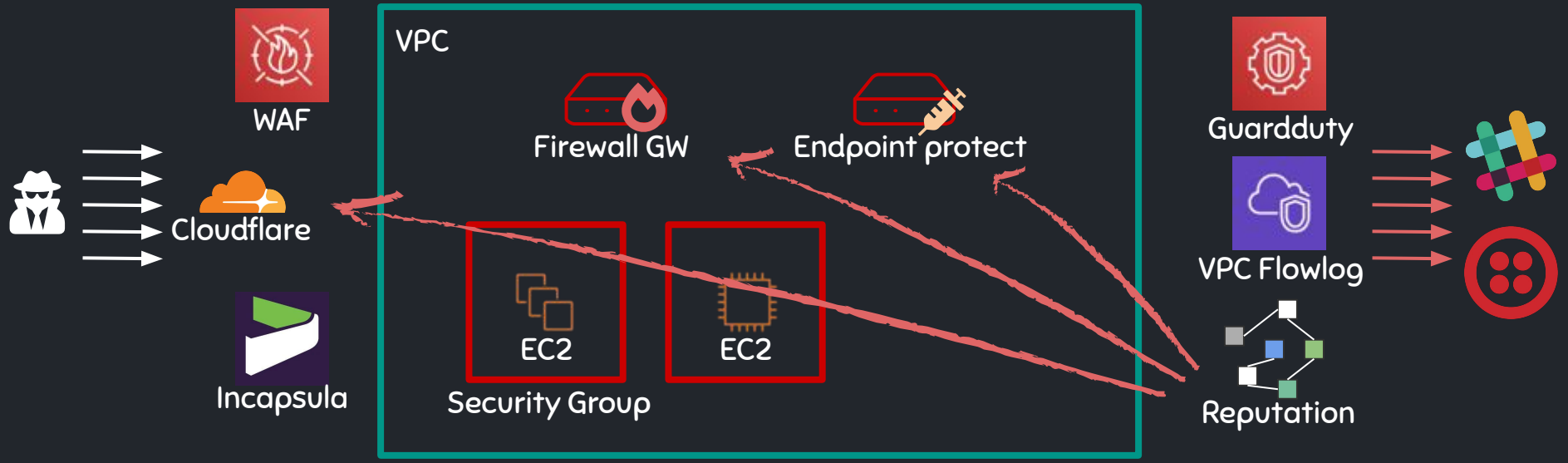


發生維運人員或駭客存取時...

例如某個IAM User使用了他平常不用的API
或是KMS裡面的加密金鑰在異常的時間或地點被存取

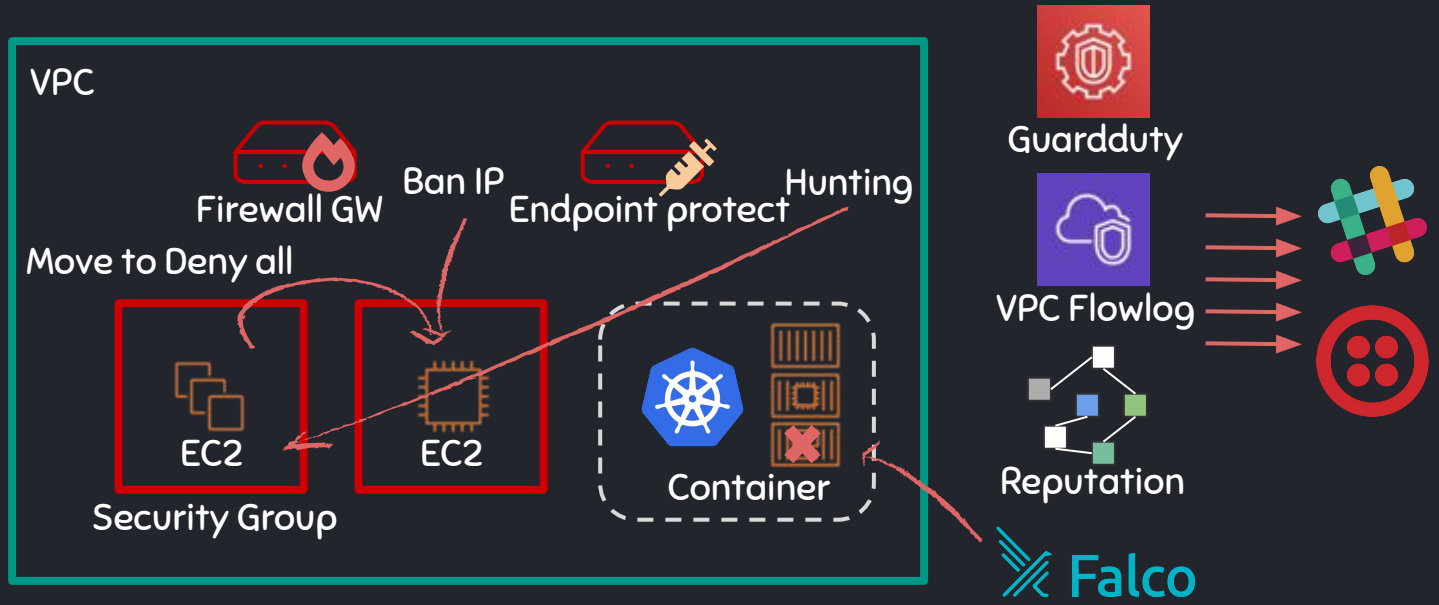
惡意攻擊

發生從外部發起的攻擊時...



惡意攻擊

發現從內部發起的攻擊時...



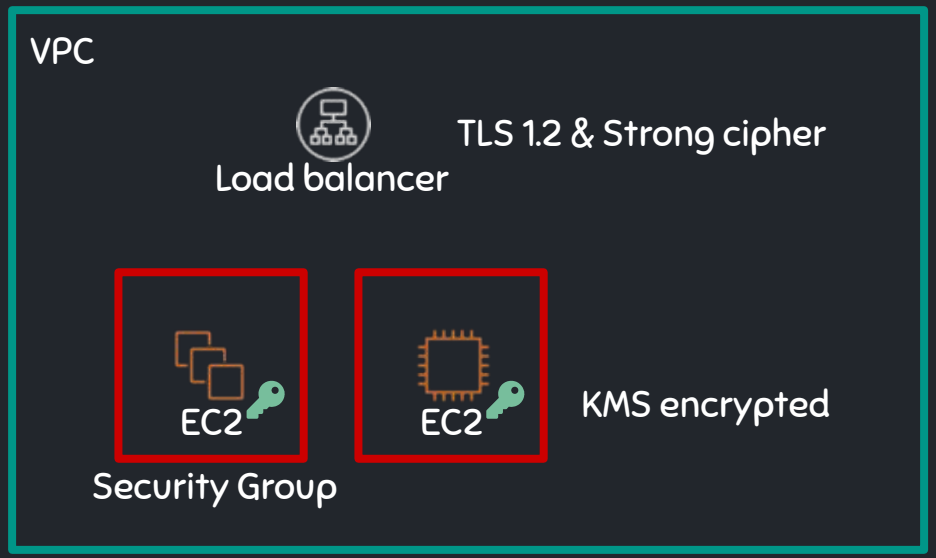
Ref: https://aws.amazon.com/tw/blogs/opensource/securing-amazon-eks-lambda-falco/
https://sysdig.com/blog/how-to-identify-malicious-ip-activity-using-falco/

PCI REQUIREMENT

保持系統符合PCI的要求，例如:

- SSL Cipher policy
- Storage encryption

KMS

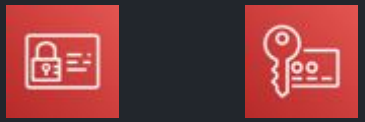


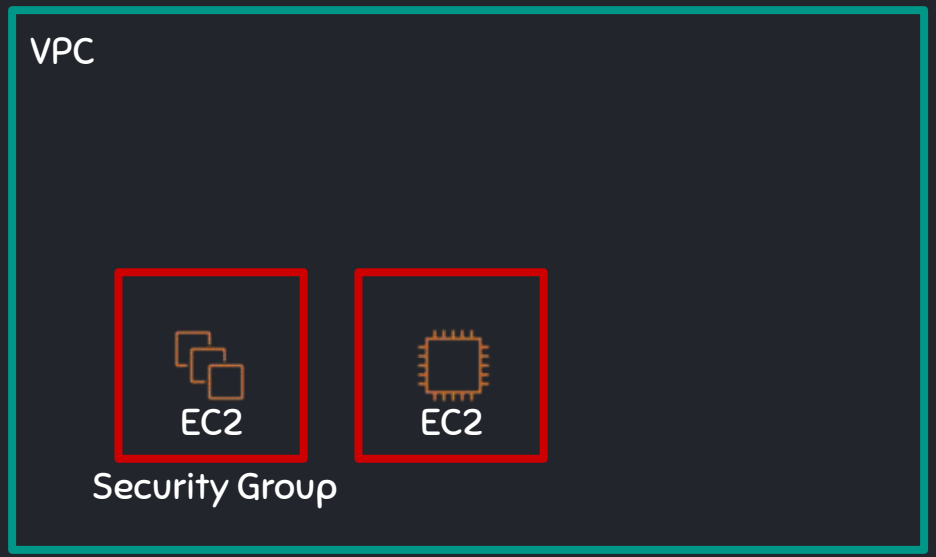
Config

CIS CONTROL

保持系統符合CIS的要求,
例如:

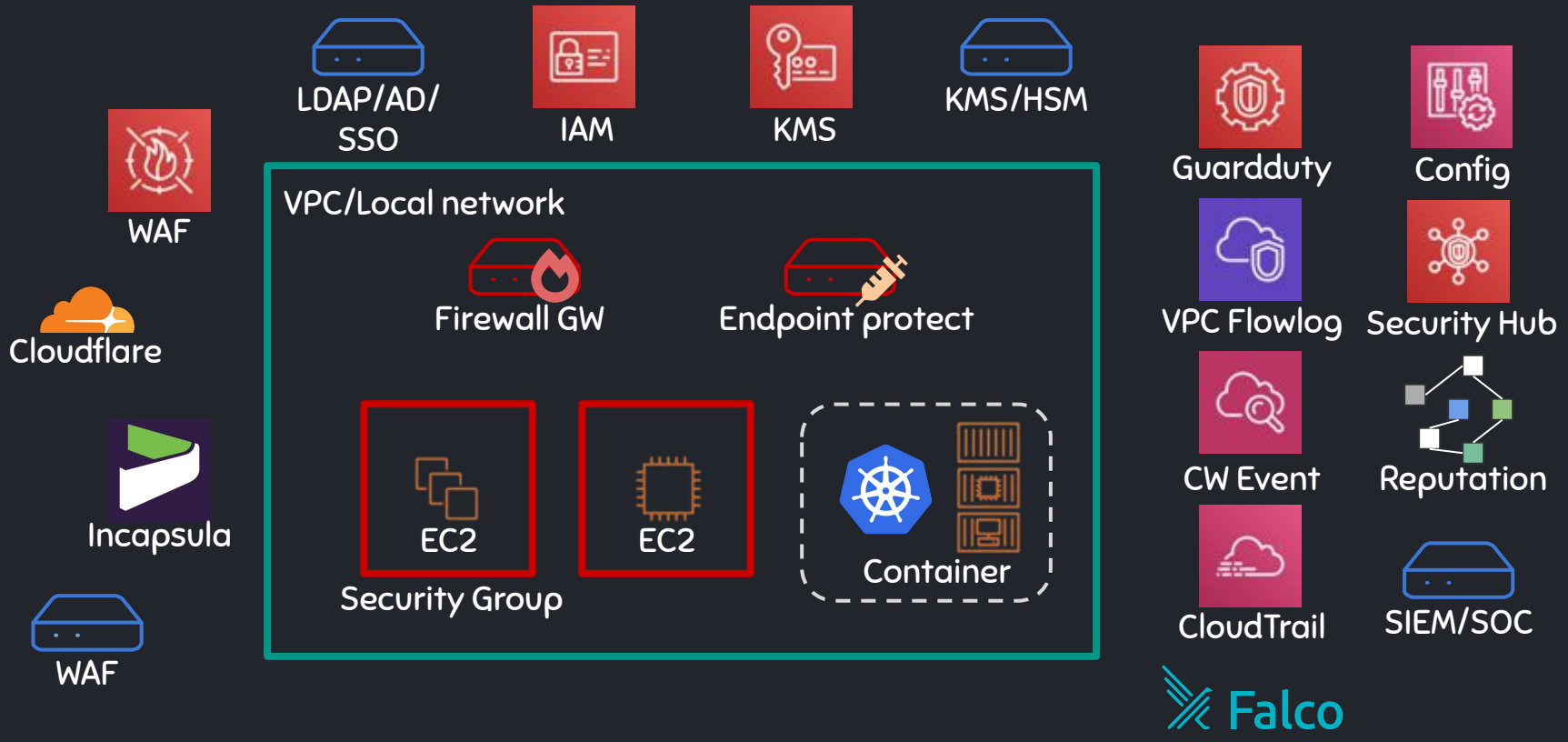
- IAM User變更警告
- 必須使用MFA
- 網路架構變更警告
- KMS變更警告
- 未授權API使用警告





https://github.com/4ndersonLin/AWS-CIS-alert

Overview Again



Conclusions

Conclusions
Back to the requirement

我們是否真的需要DevSecOps?

軟性需求、剛性需求

有沒有我們可以立刻開始做的?

CI/CD with Security Testing, IaC

我們要做到什麼成熟度?

PaC, Automation response

Conclusions

最痛苦的地方是...

開源工具不夠成熟

較為成熟的工具又有限定語言或是限定vendor的問題

THANKS!

Any questions?

You can find me at
security@maicoin.com

We are Hiring

Software Engineer in Test

Software Engineer

https://github.com/MaiAmis/Careers