Security Testing

YSc

Computer Security 2020/01/03

自動化

Q & A

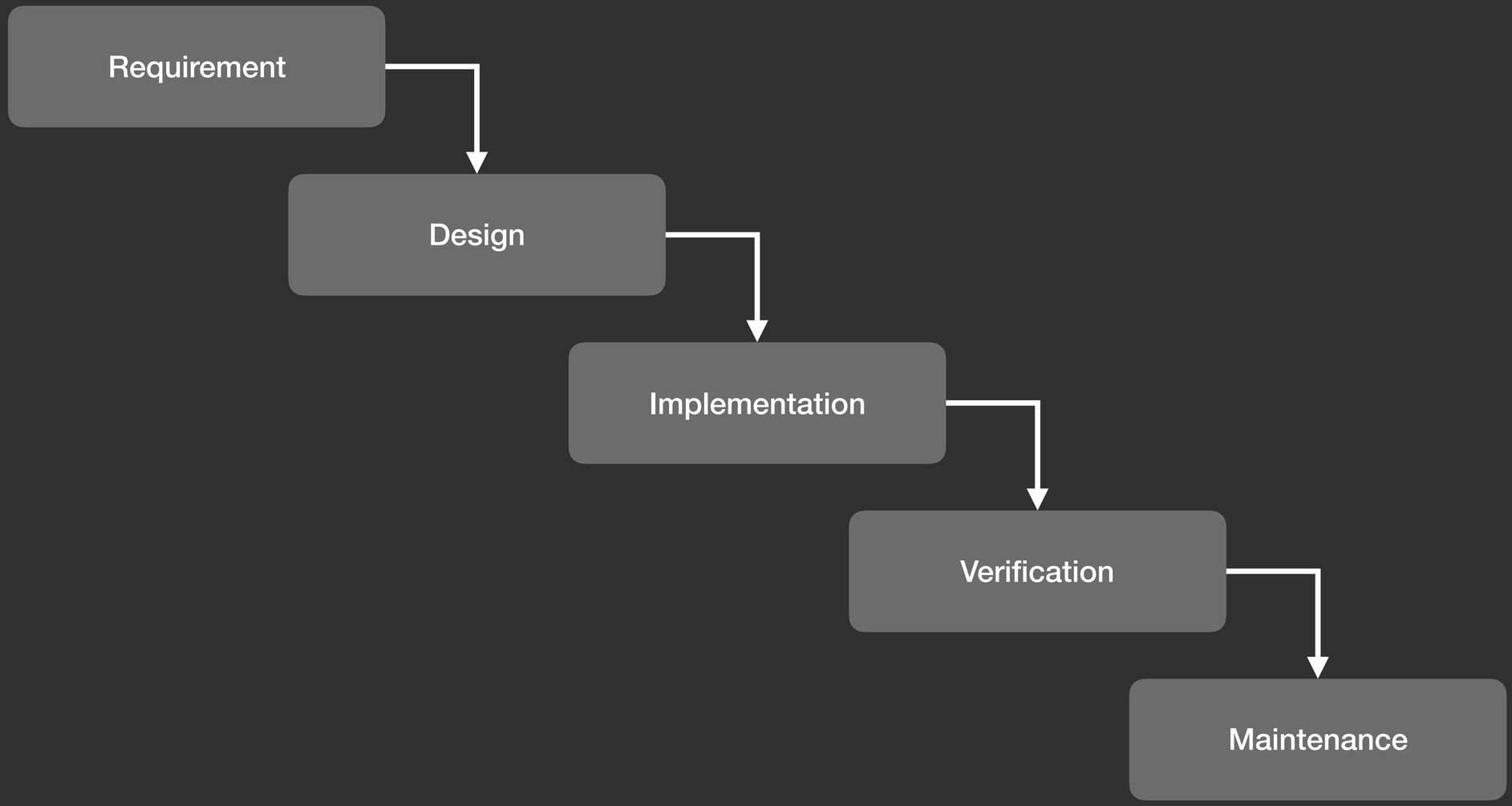
YSc

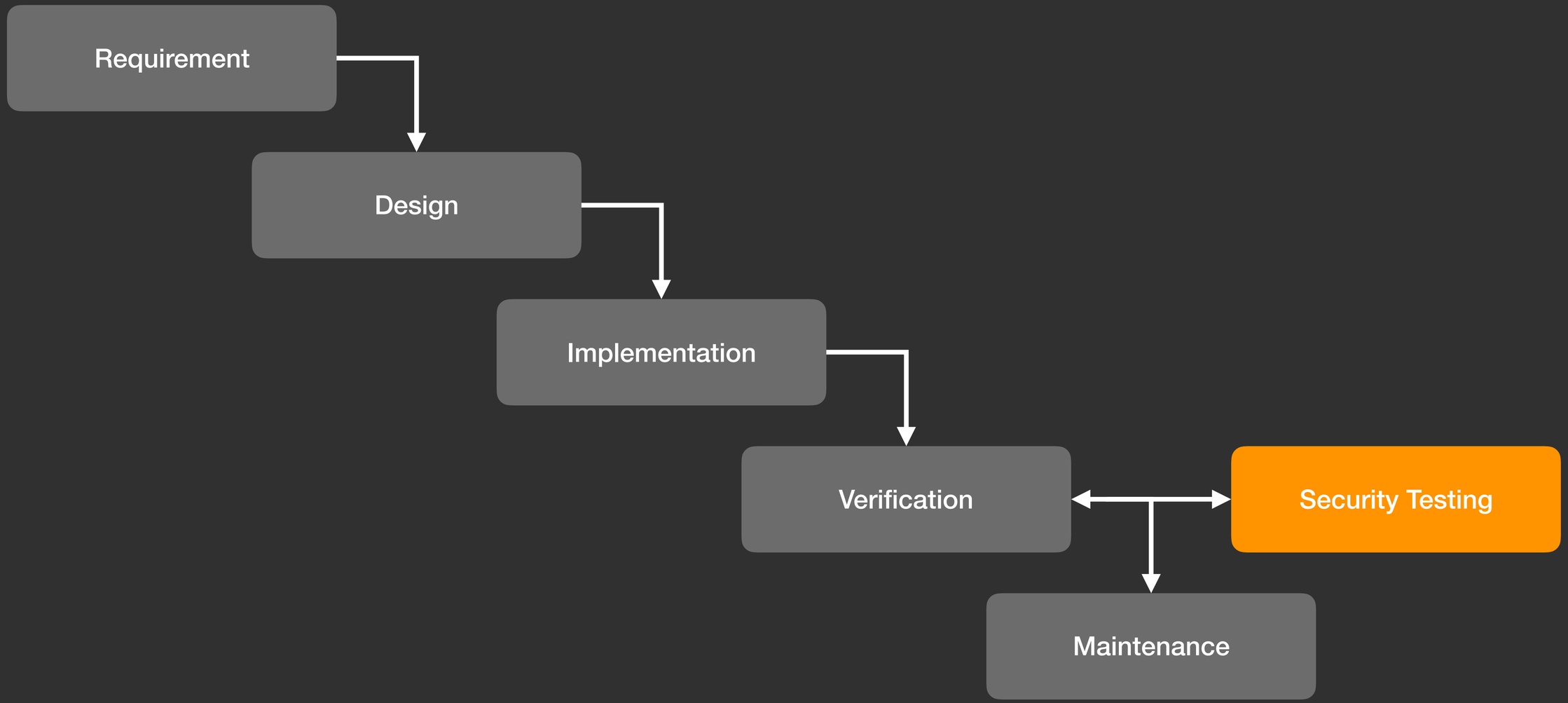
- Balsn CTF Team Co-Founder
- Bug Bounty Hunter
- HITCON CMT & Defense & Training Speaker
- Modern Web & DevOps Days Speaker
- 白帽觀點 https://secview.io/

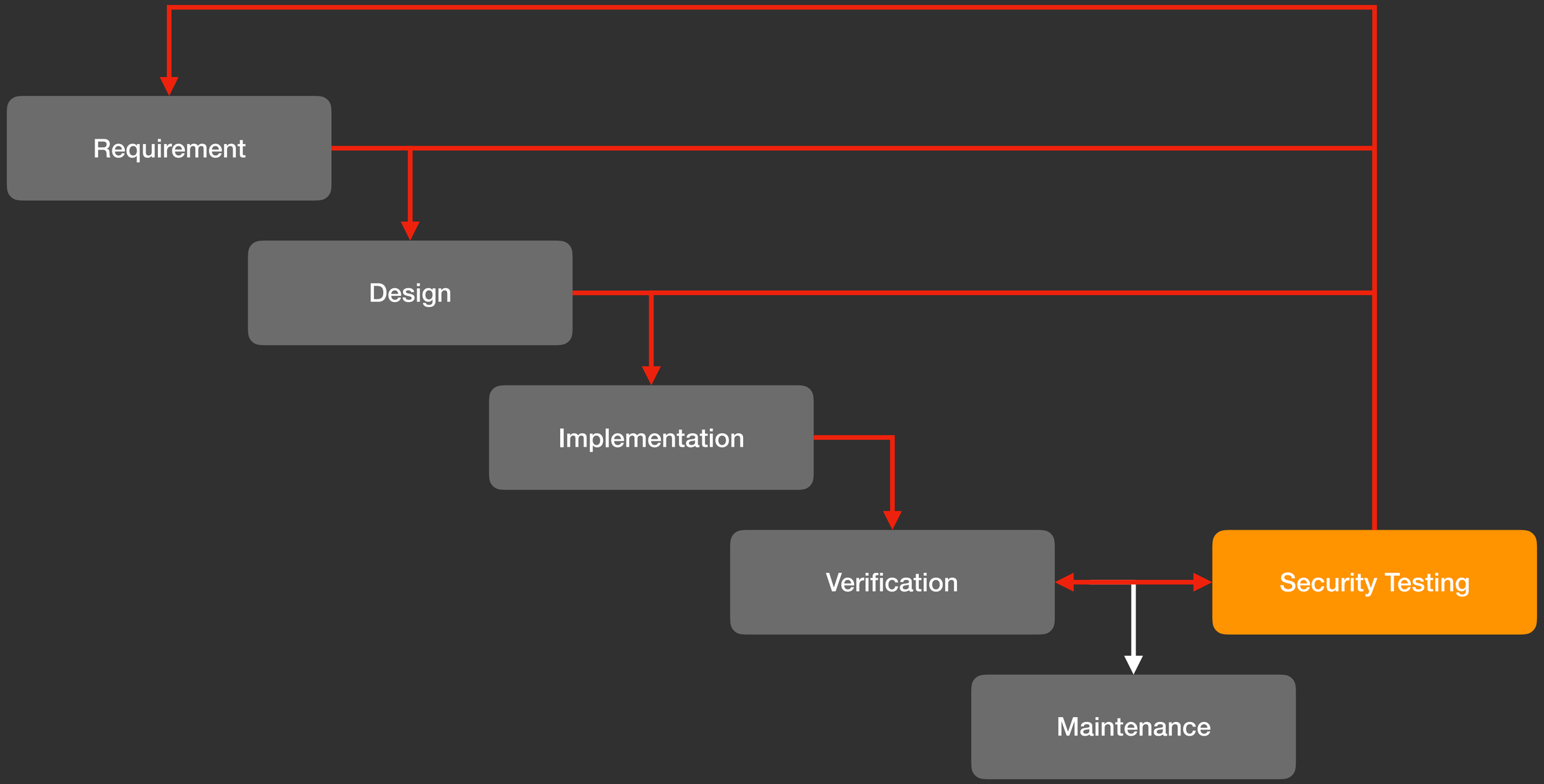
課程大綱

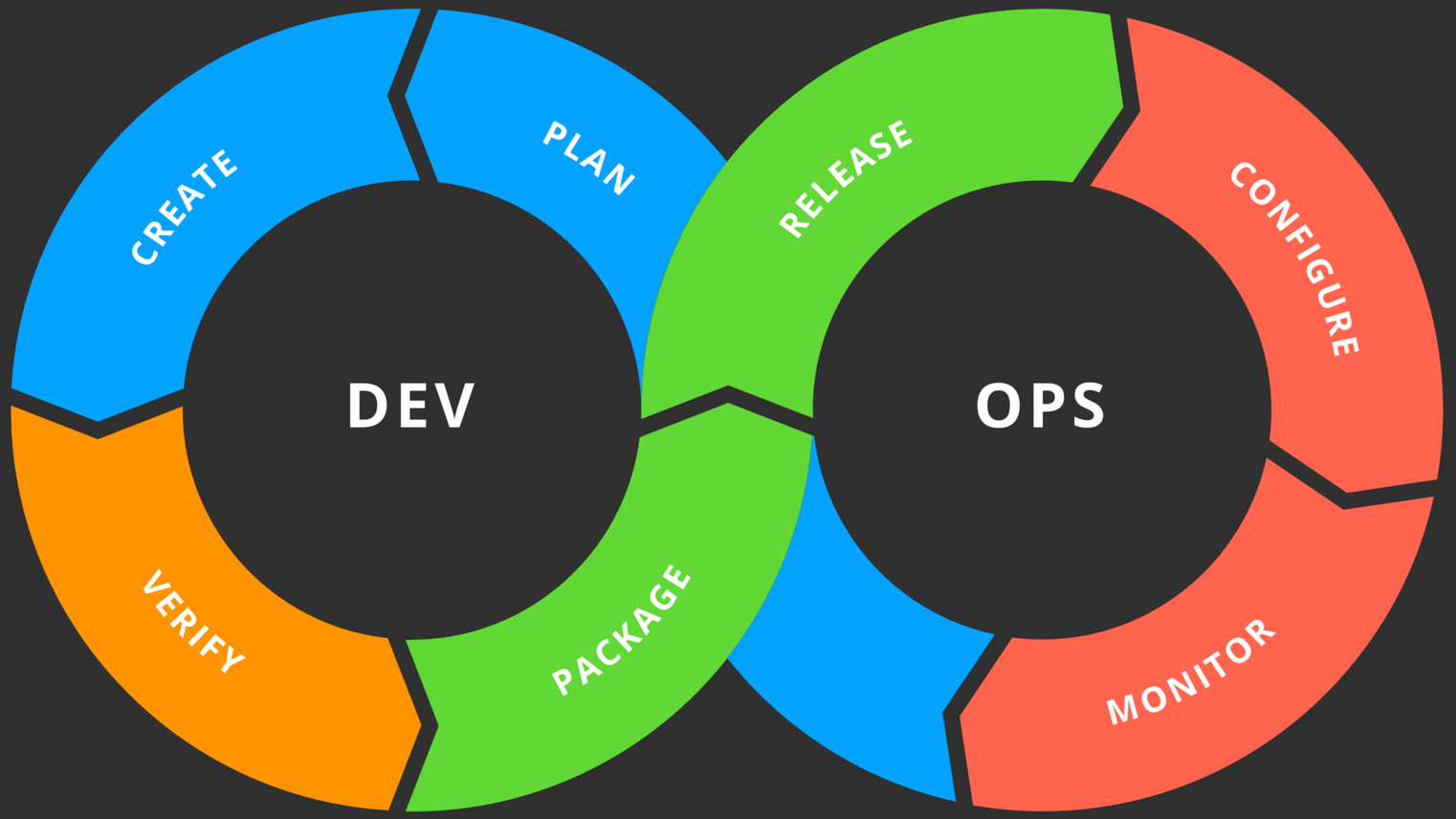
- DevOps & Security
- Security Testing

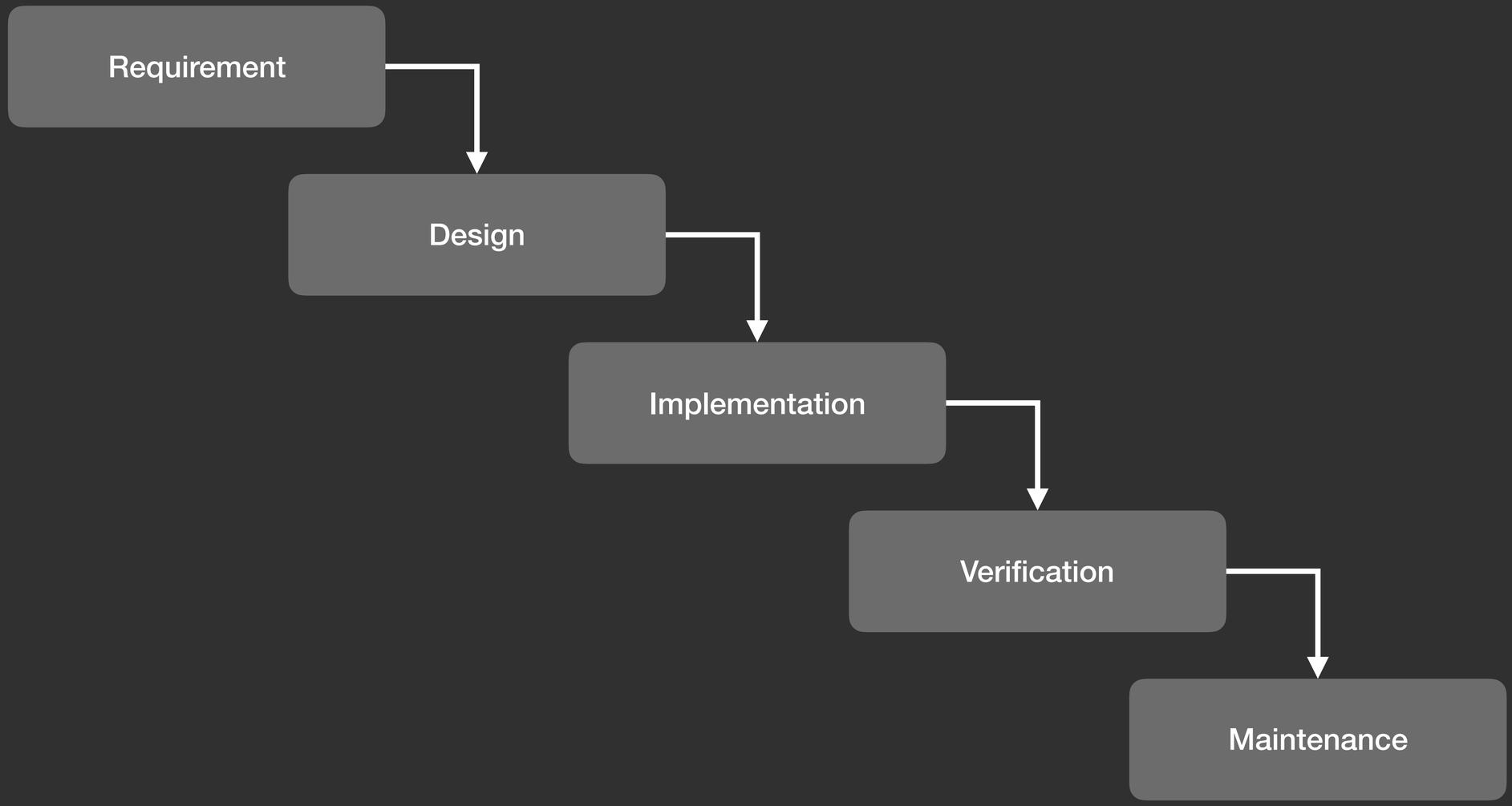
DevOps & Security

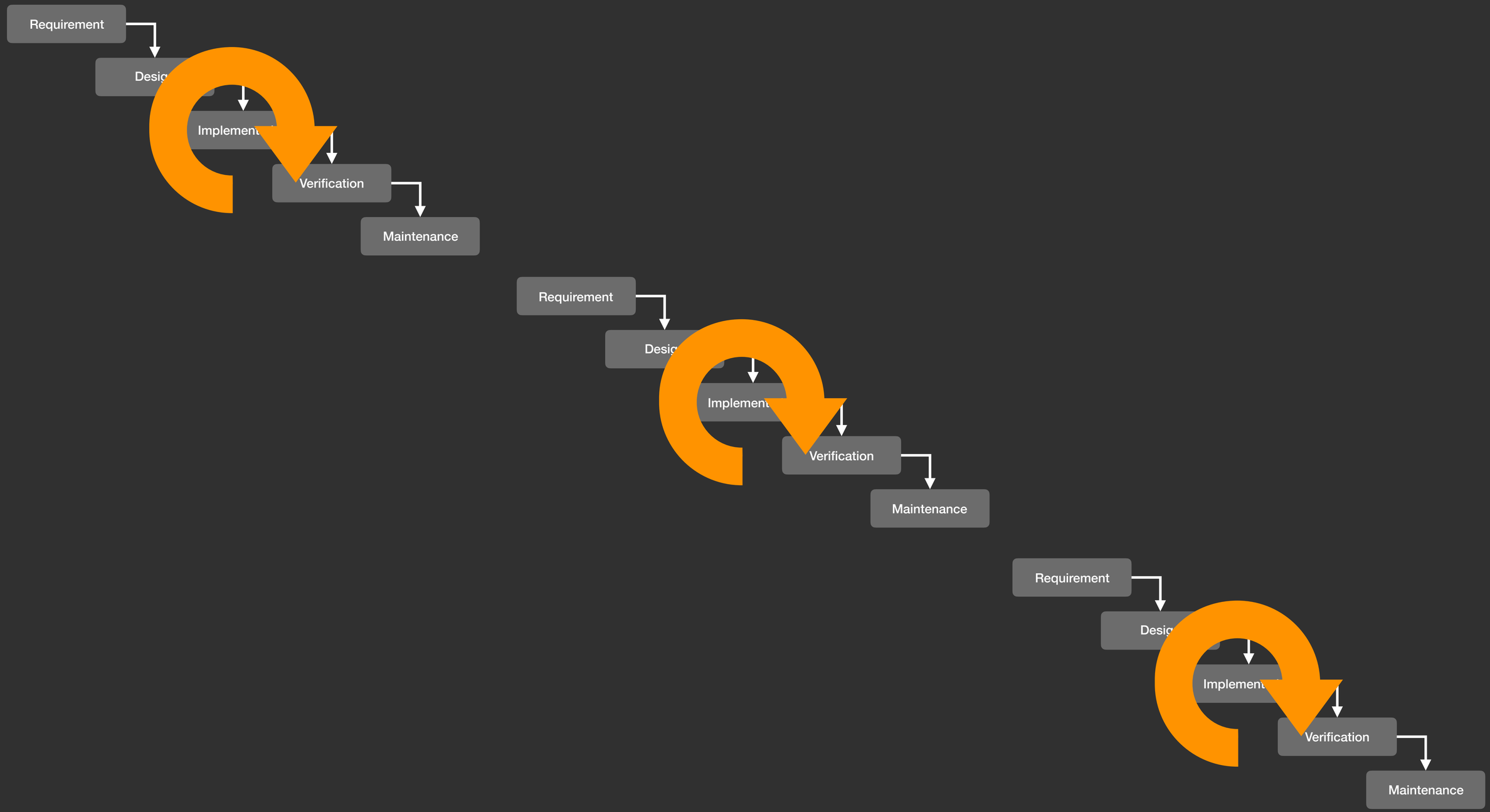

Requirement
Design
Implementation
Verification
Maintenance

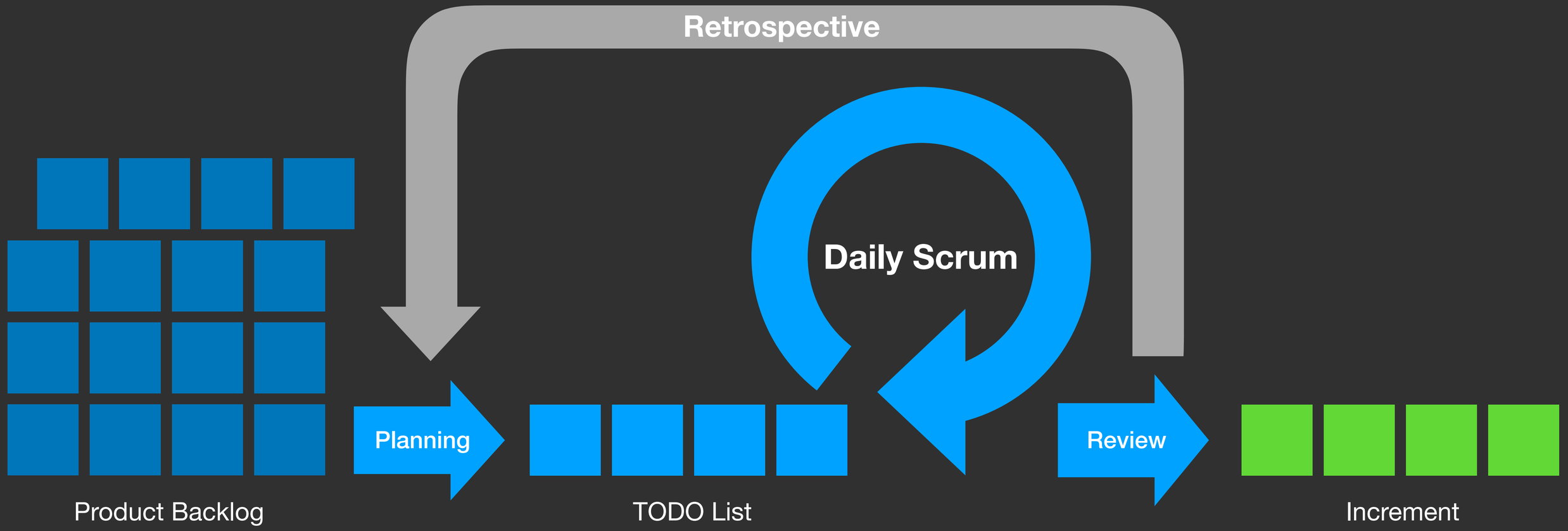

Requirement
Design
Implementation
Verification
Security Testing
Maintenance


Requirement
Design
Implementation
Verification
Security Testing
Maintenance


CREATE
PLAN
DEV
VERIFY
PACKAGE
RELEASE
CONFIGURE
OPS
MONITOR


Requirement
Design
Implementation
Verification
Maintenance


Requirement
Desig
Implement
Verification
Maintenance
Requirement
Desig
Implement
Verification
Maintenance
Requirement
Desig
Implement
Verification
Maintenance


Retrospective
Daily Scrum
Planning
Review
Product Backlog
TODO List
Increment

Development

Search or filter results...

Show labels

View scope

Open

123 0

Add support for a moduleless (single module) docs component

[Domain] Backend [Progress] In Prototype [Type] Feature

#27

Module paths should be configurable

[Domain] Backend [Type] Feature

#28

Add option to playbook to skip worktree(s)

[Domain] Backend [Type] Feature

#82

Separate content aggregator from git provider

[Domain] Backend [Type] Feature

#93

Gather UI acceptance test suite requirements

[Domain] Build / CI [Type] Decision

#95

Ignore duplicate component in same repository if matches component in HEAD

[Domain] Backend [Type] Bug

#120

Decide whether content aggregate should be sorted

[Domain] Architecture [Type] Design

#121

[Progress] On Hold

0 0

[Progress] Blocked

0 0

[Progress] Preparing Change

4 0

[Progress] In Review

15 0

Add (Apache) httpd redirect facility to redirect producer

[Domain] Backend [Type] Feature

#192

Docs improvement: Quick start guide

[Domain] Documentation [Type] Improvement

#299

Automatically register redirect for start page of component version

[Domain] Backend [Type] Feature

#379

Page version selector and canonical URL should take page aliases into account

[Domain] Backend [Type] Feature

#425

Document how to create a duplicate entry in the navigation that's independently selectable

[Domain] Documentation

#427

Document implicit hidden and non-publishable pages

[Domain] Documentation [Type] Feature

#461

Always add trailing newline to any generated files

[Domain] Backend [Type] Bug

[Progress] Change Required

1 0

Closed

Initiate default UI project

[Domain] Project Mgmt [Type] Task

#20

Set up job to run tests in GitLab CI

[Domain] Build / CI [Type] Task

#6

Architect the playbook software com

[Domain] Architecture [Domain] Docum

[Type] Design

#10

Document playbook architecture dec

[Domain] Architecture [Domain] Docum

[Type] Design

#15

Configure coding style rules

[Domain] Backend [Type] Task

#5

Specify code contribution workflow

[Domain] Documentation [Type] Decisio

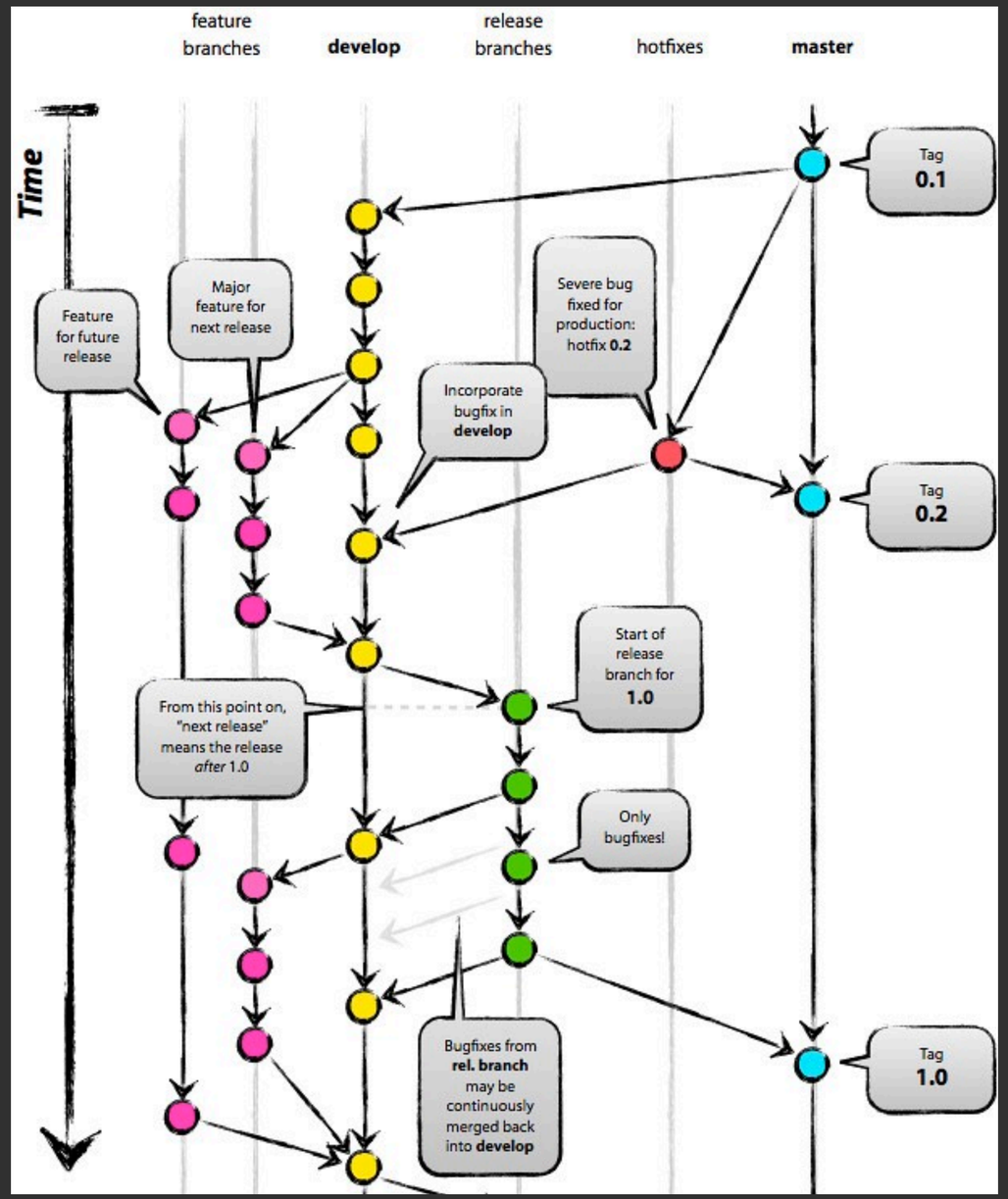
#13

Initiate README document

[Domain] Documentation [Type] Task

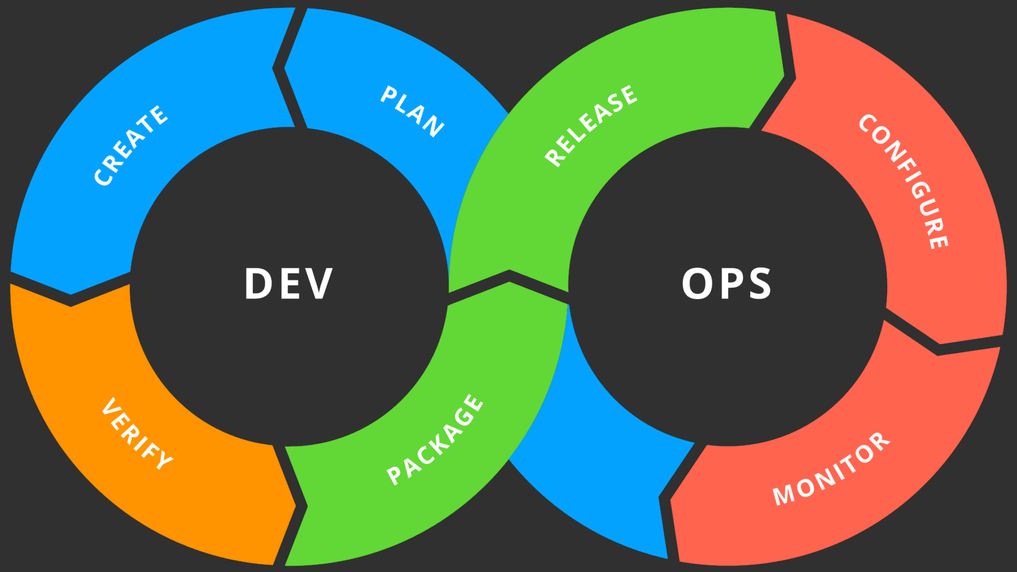
#8

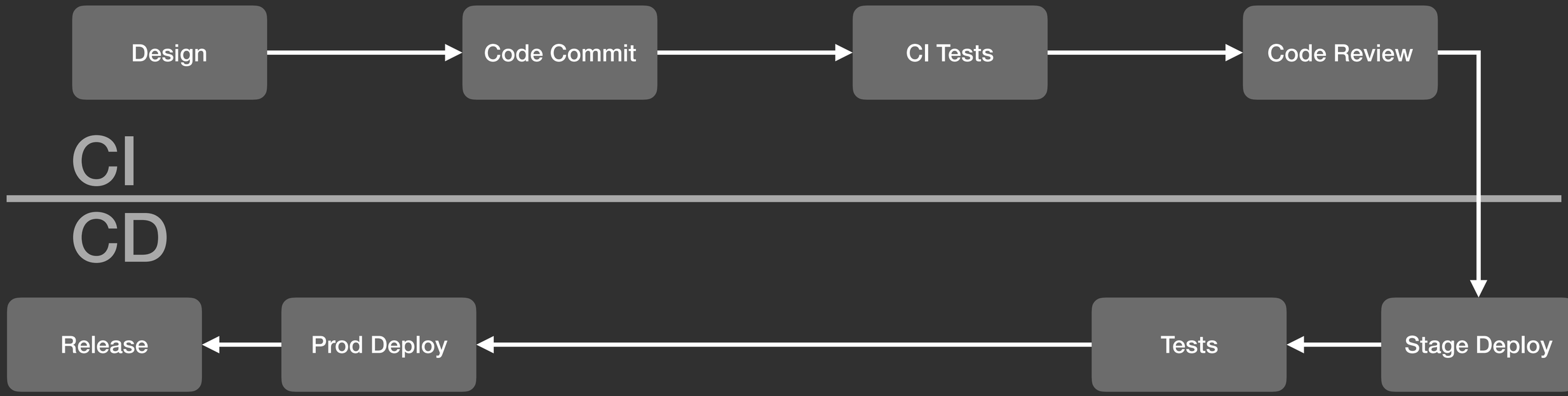
Git Flow

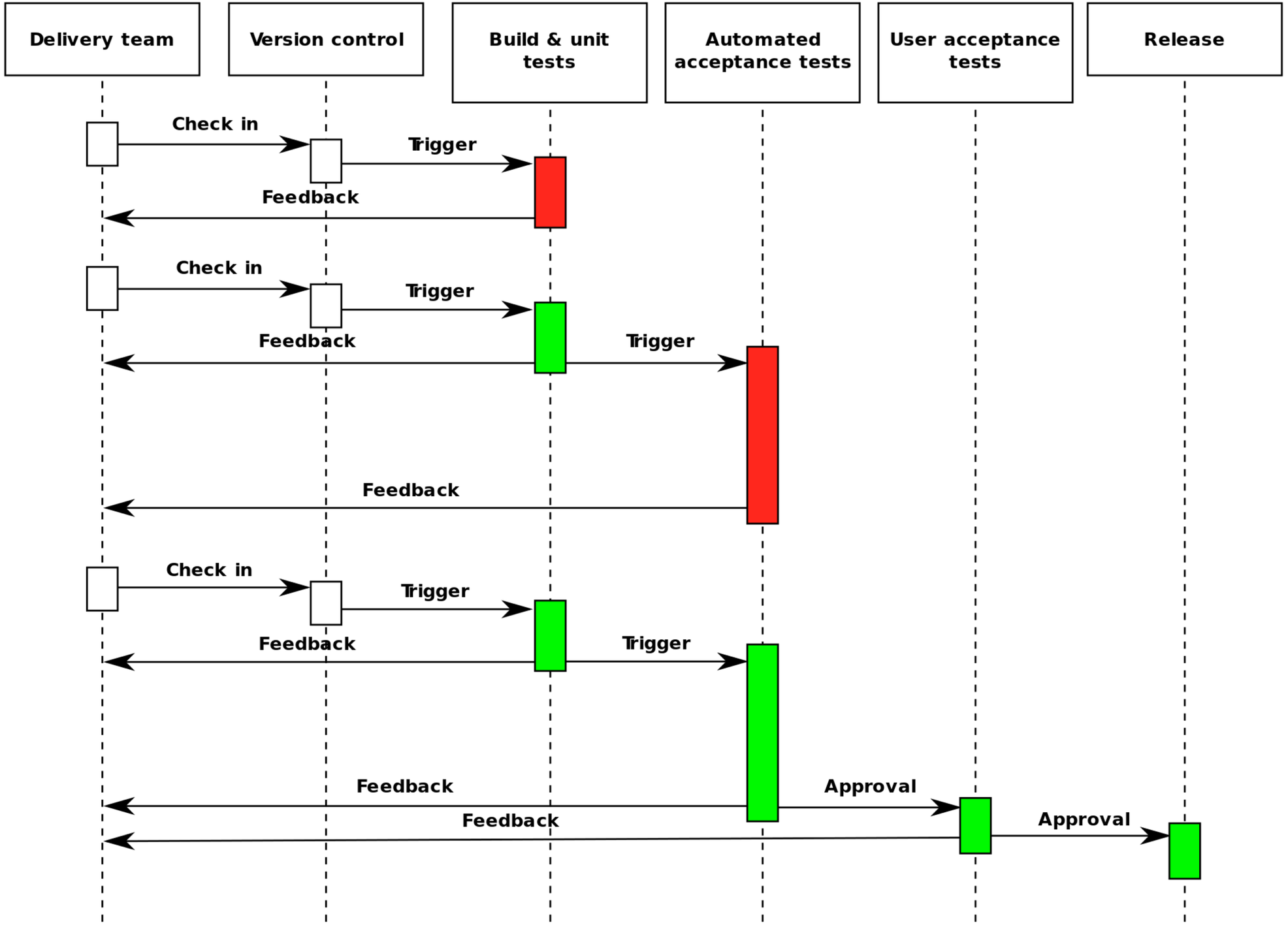

feature branches
develop
release branches
hotfixes
master
Time
Tag 0.1
Feature for future release
Major feature for next release
Severe bug fixed for production: hotfix 0.2
Incorporate bugfix in develop
Tag 0.2
Start of release branch for 1.0
From this point on, "next release" means the release after 1.0
Only bugfixes!
Bugfixes from rel. branch may be continuously merged back into develop
Tag 1.0

CI / CD

- 持續整合（Continuous Integration）
  - 將新開發的程式碼整合進主線（e.g. release branch）前的自動化檢查
  - 單元測試、格式檢查等等，為了解決錯誤、改善軟體品質
- 持續交付（Continuous Delivery）
  - 為了可以快速開發，且保證軟體可以穩定、持續的保持在隨時可以釋出的狀況
  - 將主線上的程式碼部署到伺服器上的自動化腳本

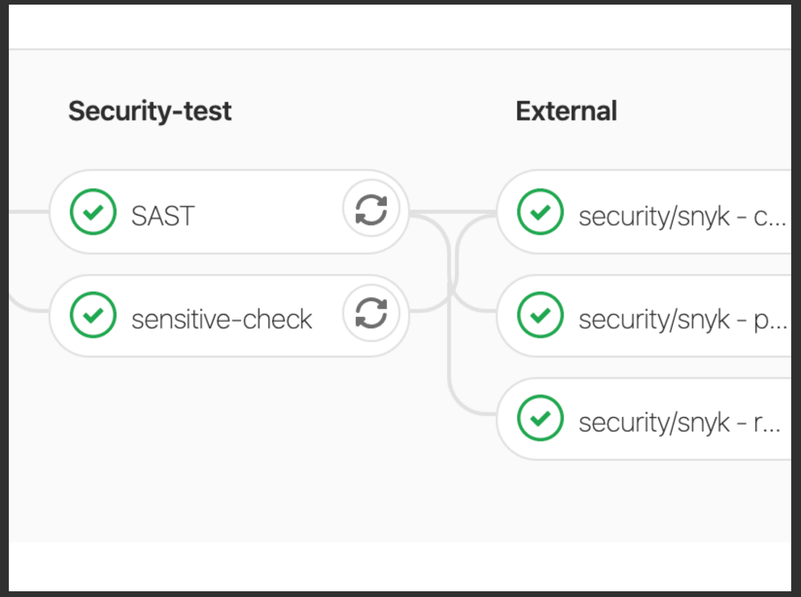


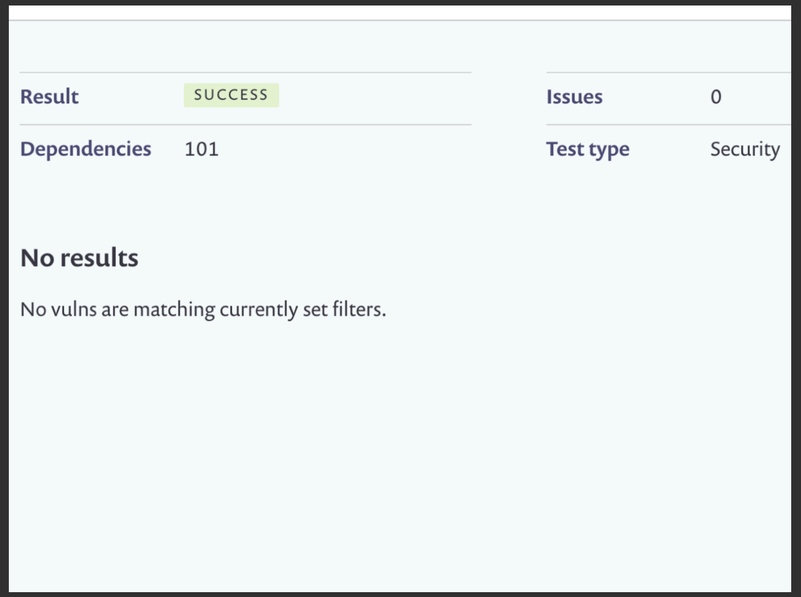

Design
Code Commit
CI Tests
Code Review
CI
CD
Release
Prod Deploy
Tests
Stage Deploy


Delivery team
Version control
Build & unit tests
Automated acceptance tests
User acceptance tests
Release
Check in
Trigger
Feedback
Check in
Trigger
Feedback
Trigger
Feedback
Check in
Trigger
Feedback
Trigger
Feedback
Approval
Feedback
Approval

Contents of .gitlab-ci.yml

```yaml
stages:
  - linter
  - security-test
  - deploy

image: python:3.7.5-alpine3.9

linter:
  stage: linter
  script:
    - "/bin/sh ci/linter.sh"

sensitive-check:
  stage: security-test
  script:
    - "/bin/sh ci/sensitive-check.sh"

SAST:
  stage: security-test
  script:
    - "/bin/sh ci/sast.sh"

deploy:
  stage: deploy
  script:
```





DefectDojo / django-DefectDojo

build error

Current | Branches | Build History | Pull Requests | Build #2951 | More options

master CRON Update README.md

#2951 errored

Commit 6d56806

Branch master

Ran for 30 min 30 sec

Total time 1 hr 20 min 14 sec

Greg Anderson authored GitHub committed

12 hours ago

Build jobs | View config

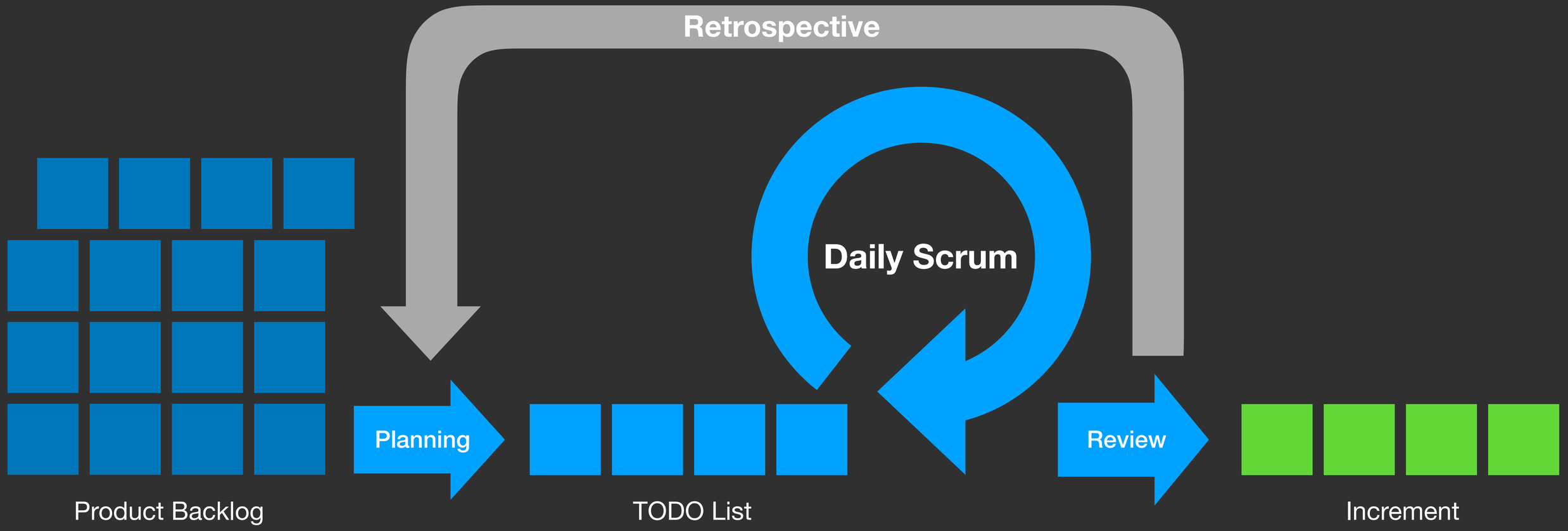
Test

14 min 8 sec

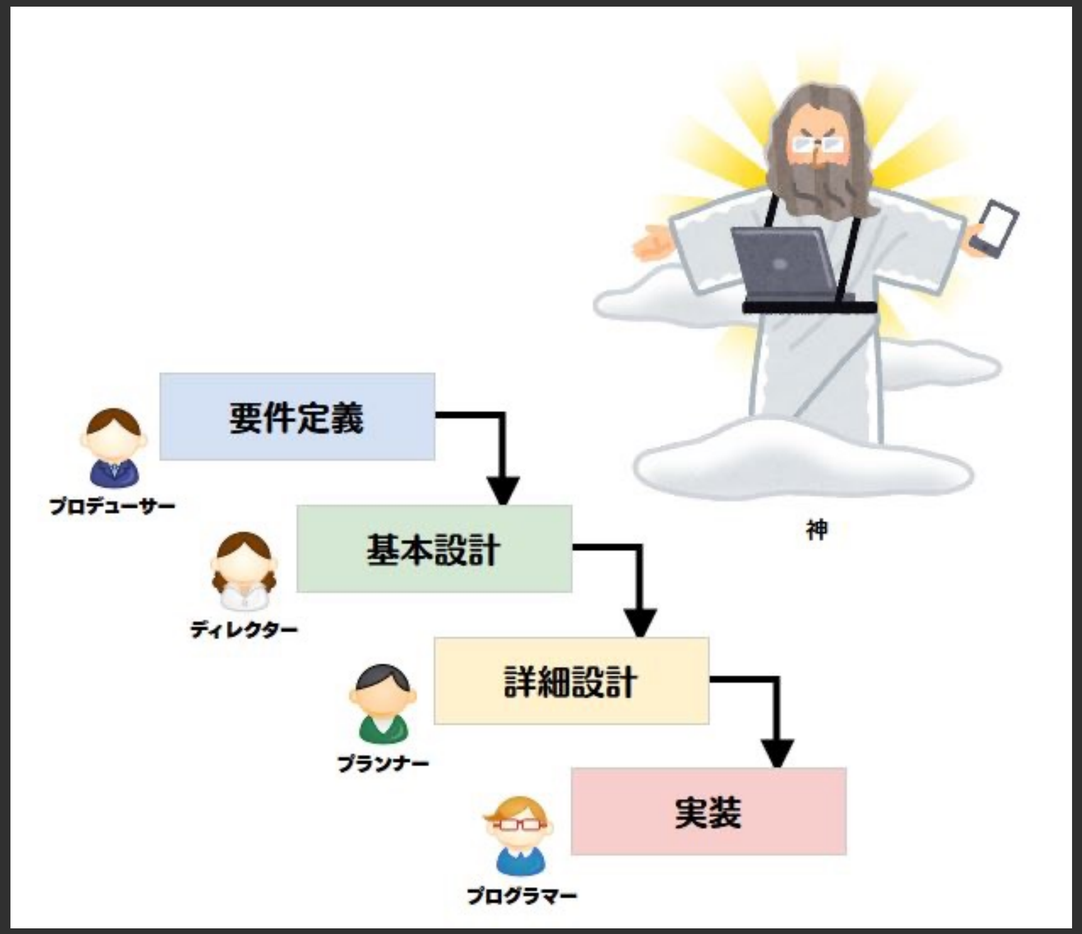
✓	# 2951.1	AMD64	no language set	BROKER=rabbitmq DATABASE=mys	12 min 35 sec
✓	# 2951.2	AMD64	no language set	BROKER=rabbitmq DATABASE=pos	12 min 27 sec
✓	# 2951.3	AMD64	no language set	BROKER=redis DATABASE=mysql	13 min 17 sec
✓	# 2951.4	AMD64	no language set	BROKER=redis DATABASE=postgre	12 min 20 sec
✓	# 2951.5	AMD64	no language set	TEST=flake8	1 min 16 sec
✓	# 2951.6	AMD64	no language set	TEST=docker	6 min 49 sec
✓	# 2951.7	AMD64	no language set	TEST=snyk	5 min 26 sec

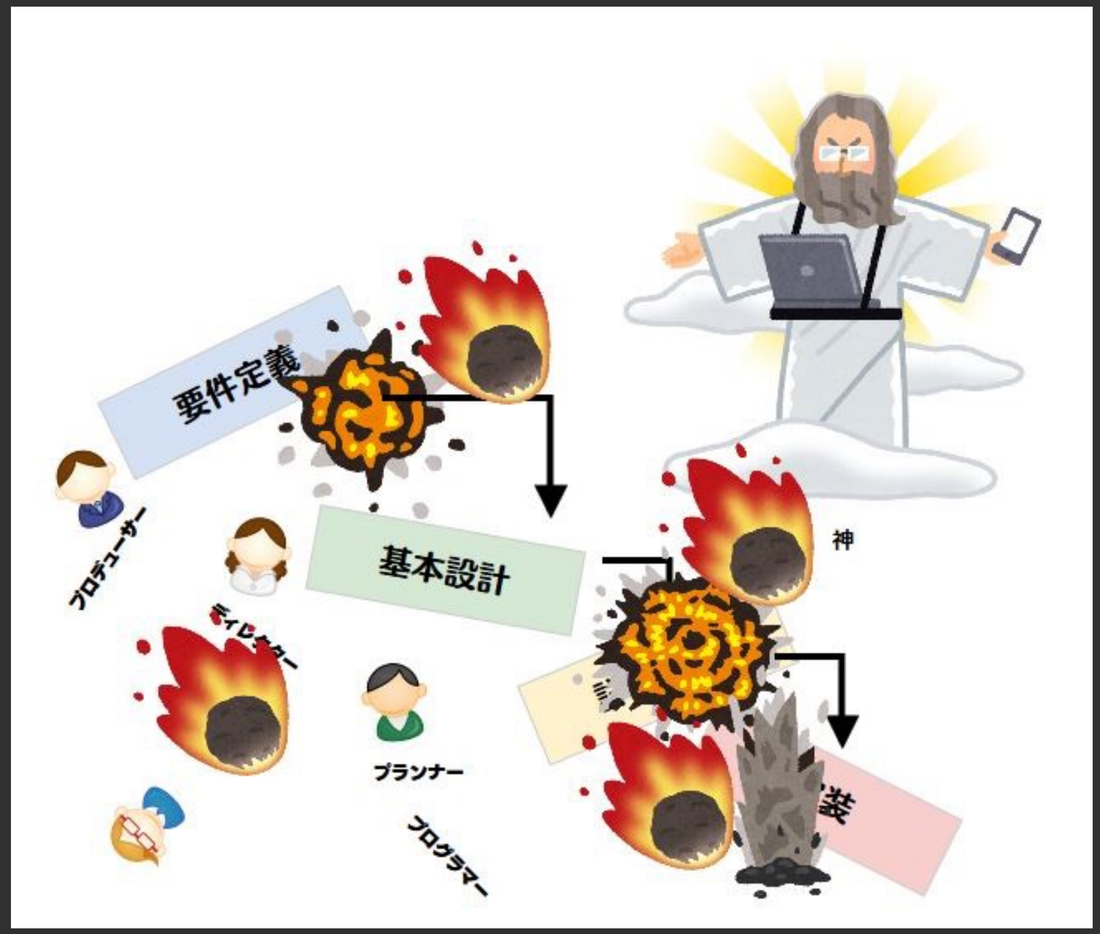
28 lines (28 sloc) | 635 Bytes

```
1  dist: xenial
2  language: minimal
3  services:
4    - docker
5  env:
6    global:
7      - K8S_VERSION=v1.13.4
8      - MINIKUBE_VERSION=v0.35.0
9      - HELM_VERSION=v2.13.0
10     - CHANGE_MINIKUBE_NONE_USER=true
11   matrix:
12     - BROKER=rabbitmq DATABASE=mysql
13     - BROKER=rabbitmq DATABASE=postgresql
14     - BROKER=redis DATABASE=mysql
15     - BROKER=redis DATABASE=postgresql
16     - TEST=flake8
17     - TEST=snyk
18     - TEST=docker
19 matrix:
20   allow_failures:
21     - env: TEST=snyk
22 jobs:
23   include:
24     - stage: deploy
25       env: TEST=deploy
26 before_install: ['./travis/before-install.sh']
27 before_script: ['./travis/before-script.sh']
28 script: ['./travis/script.sh']
```


Retrospective
Daily Scrum
Planning
Review
Product Backlog
TODO List
Increment

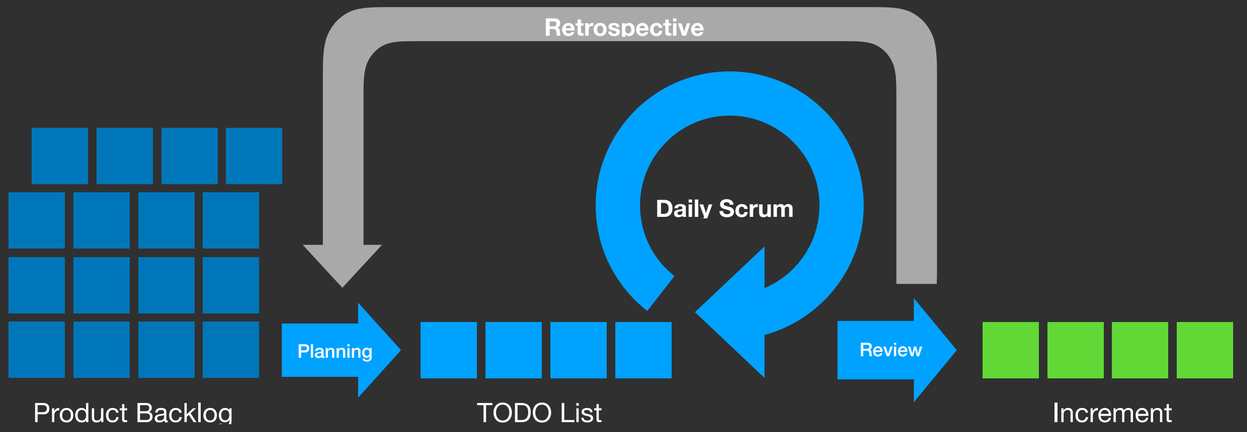
Security ?

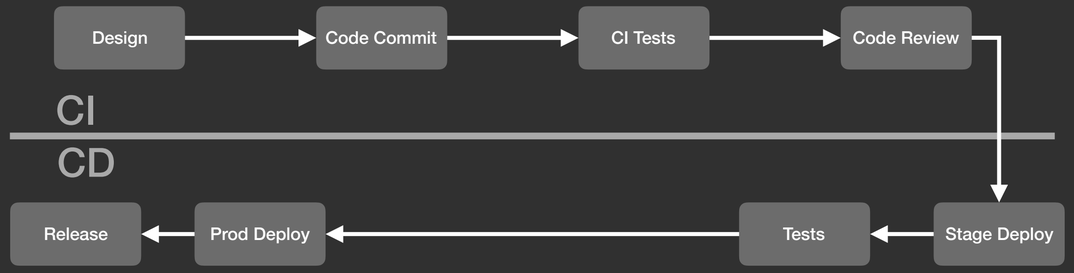

要件定義
基本設計
詳細設計
実装
プロデューサー
ディレクター
プランナー
プログラマー
神

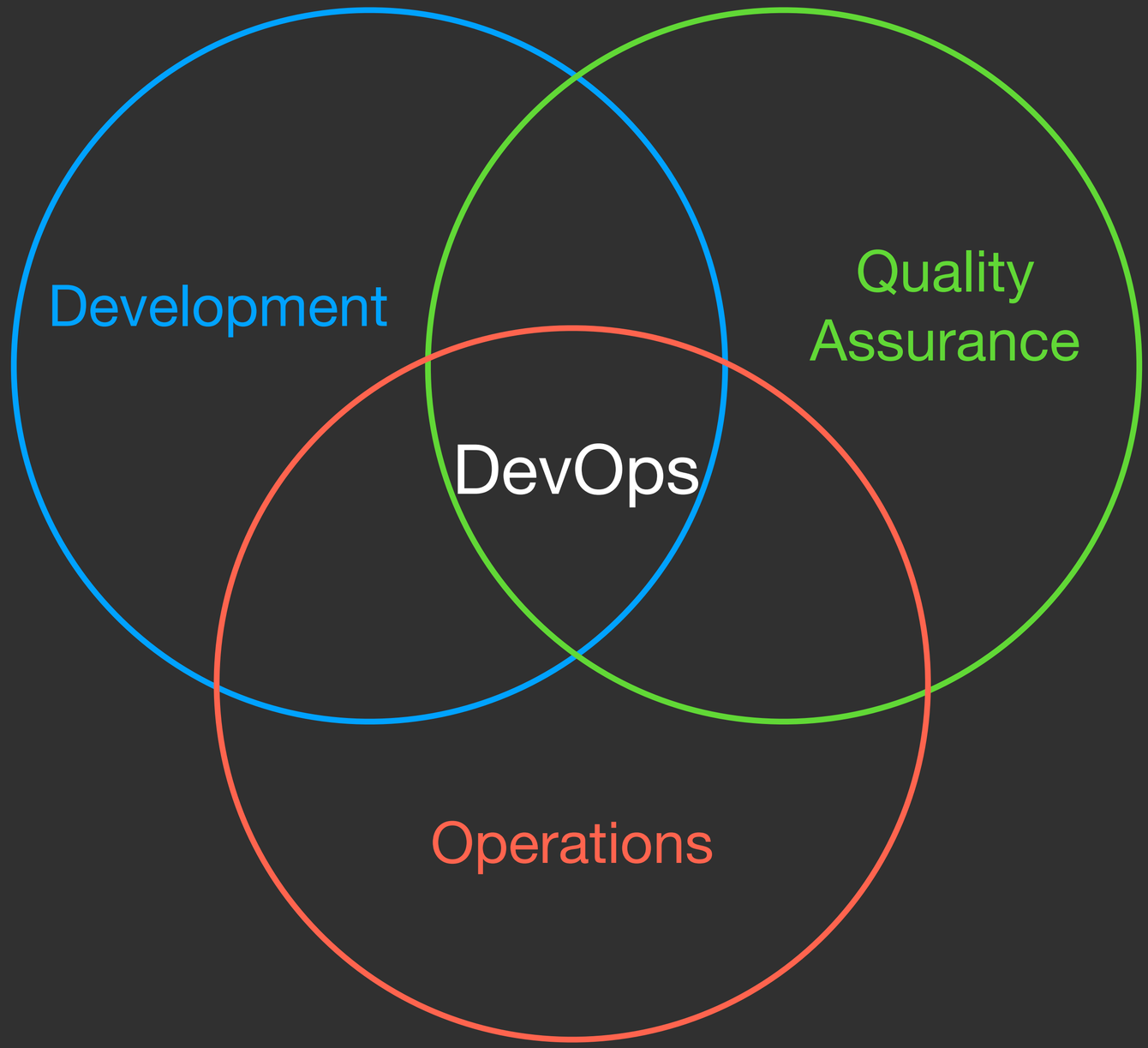

要件定義
基本設計
プロデューサー
ディレクター
プランナー
プログラマー
神

Security ?

- 持續整合（Continuous Integration）
- 持續交付（Continuous Delivery）
- 程式碼品質（Code Quality）
- 相依性組件、系統、套件（Dependencies）
- 配置（Configuration）
- 環境（Environment）
- ...






Development
Quality
Assurance
DevOps
Operations


Secure
Development
Quality & Security
Assurance
DevSecOps
Security Practice
Operations

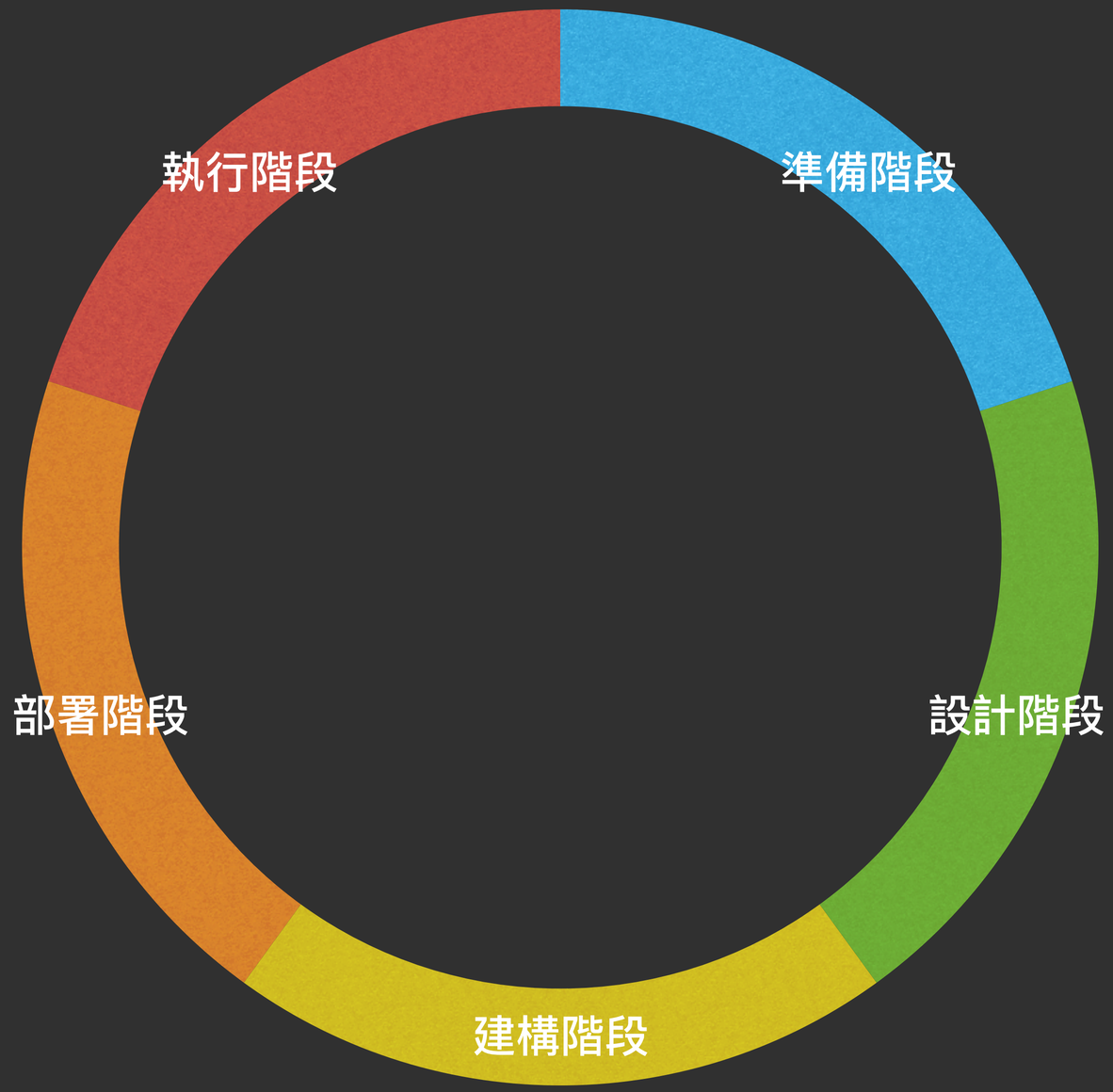
技術
（方法論）

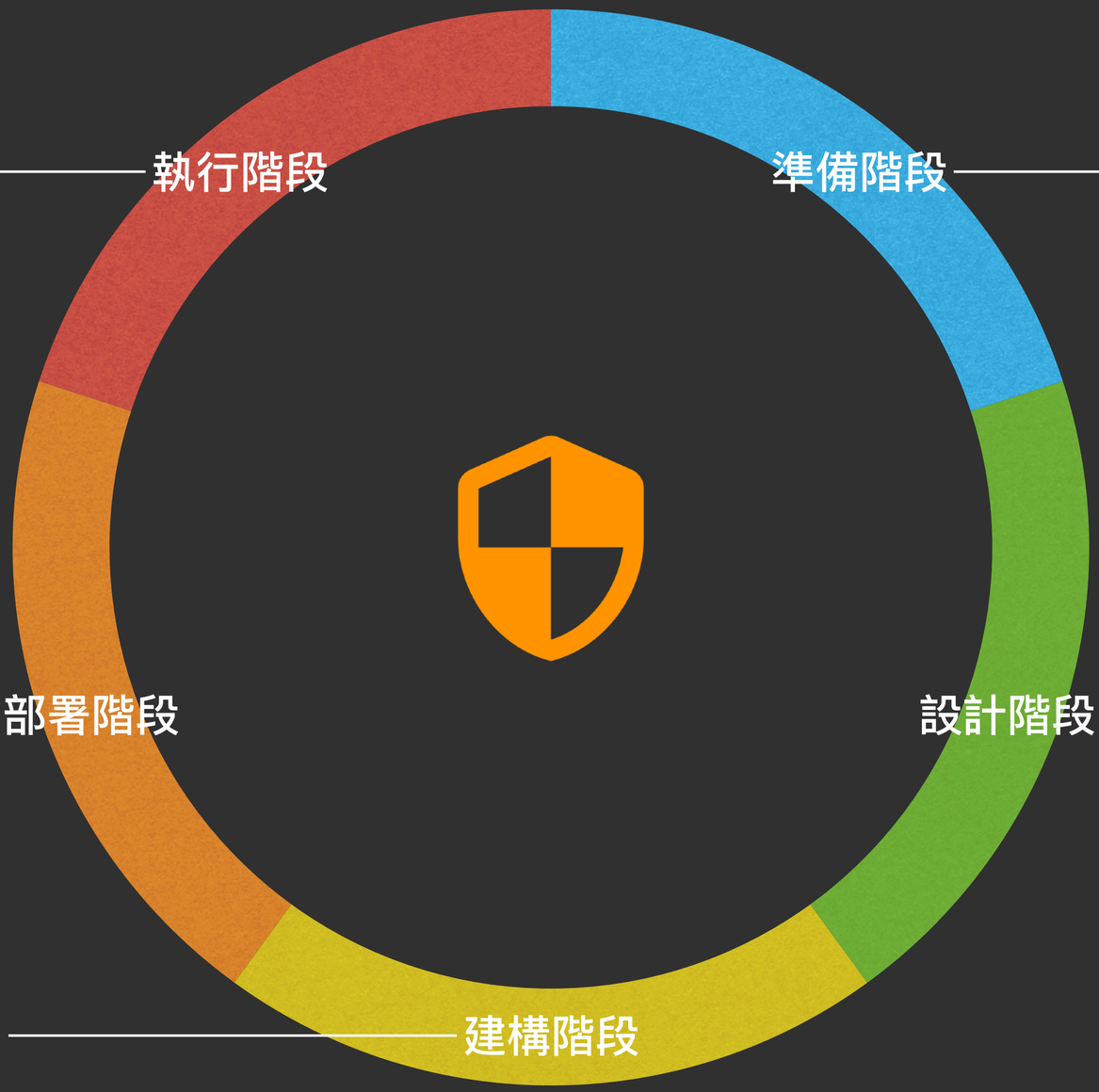
流程
（開發階段）

人
（理念）

Security is EVERYONE's job

– AWS CTO Werner Vogels


準備階段
設計階段
建構階段
部署階段
執行階段



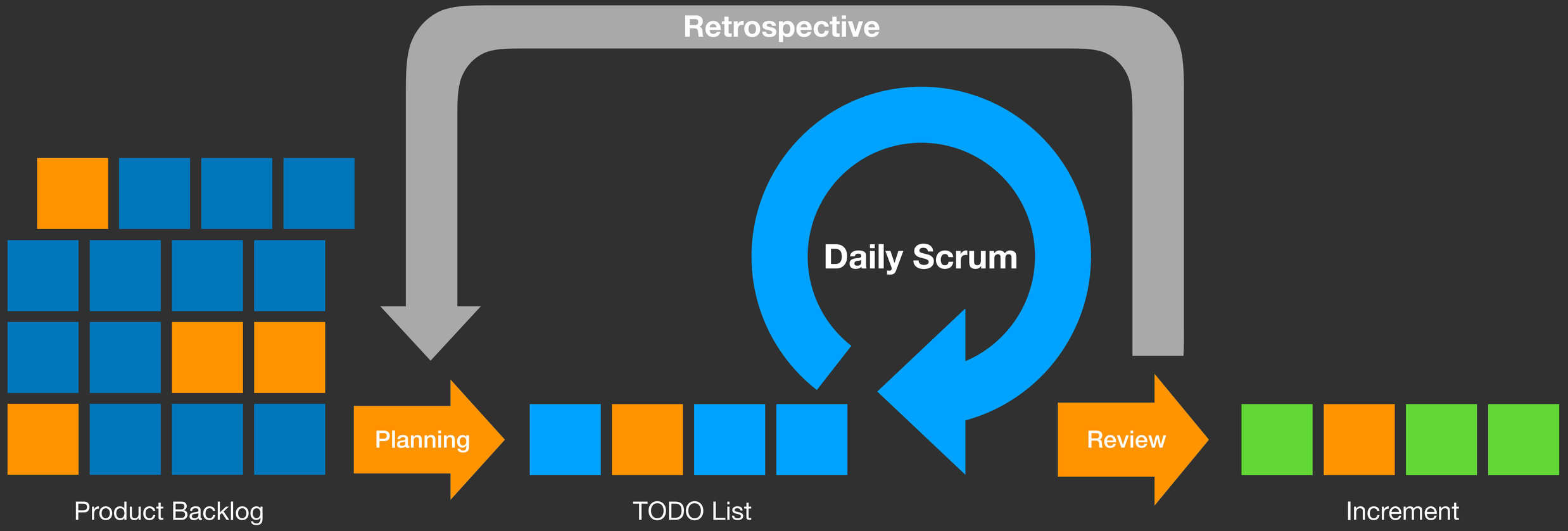
監控（Monitor）、運行時自我保護（RASP）、威脅偵測（Threat Detection）與反應（Response）

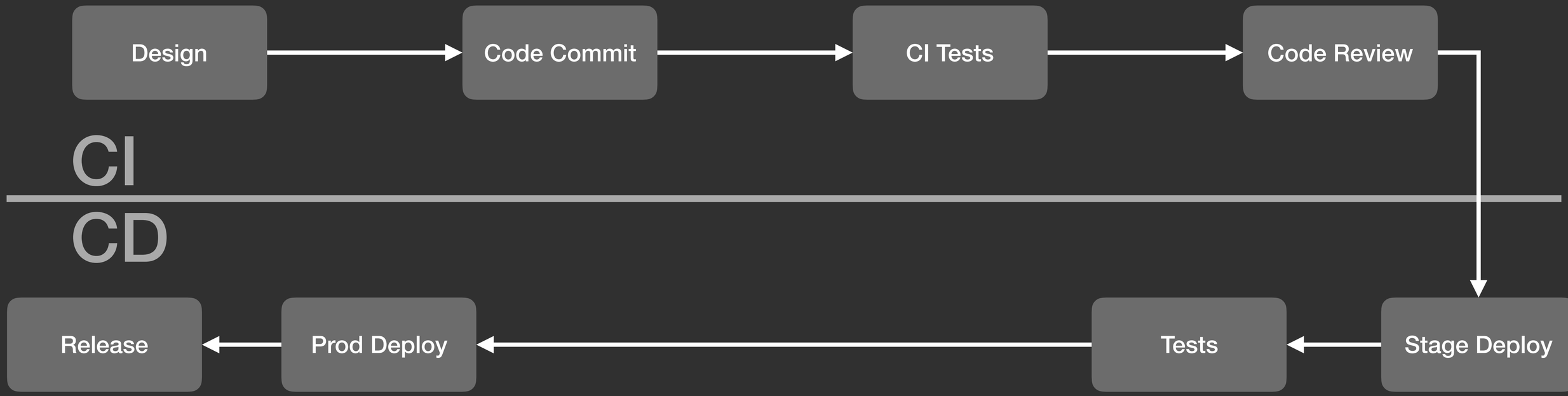
資產（Assets）管理、威脅模型（Threat Model）、風險評估（Risk Assessment）

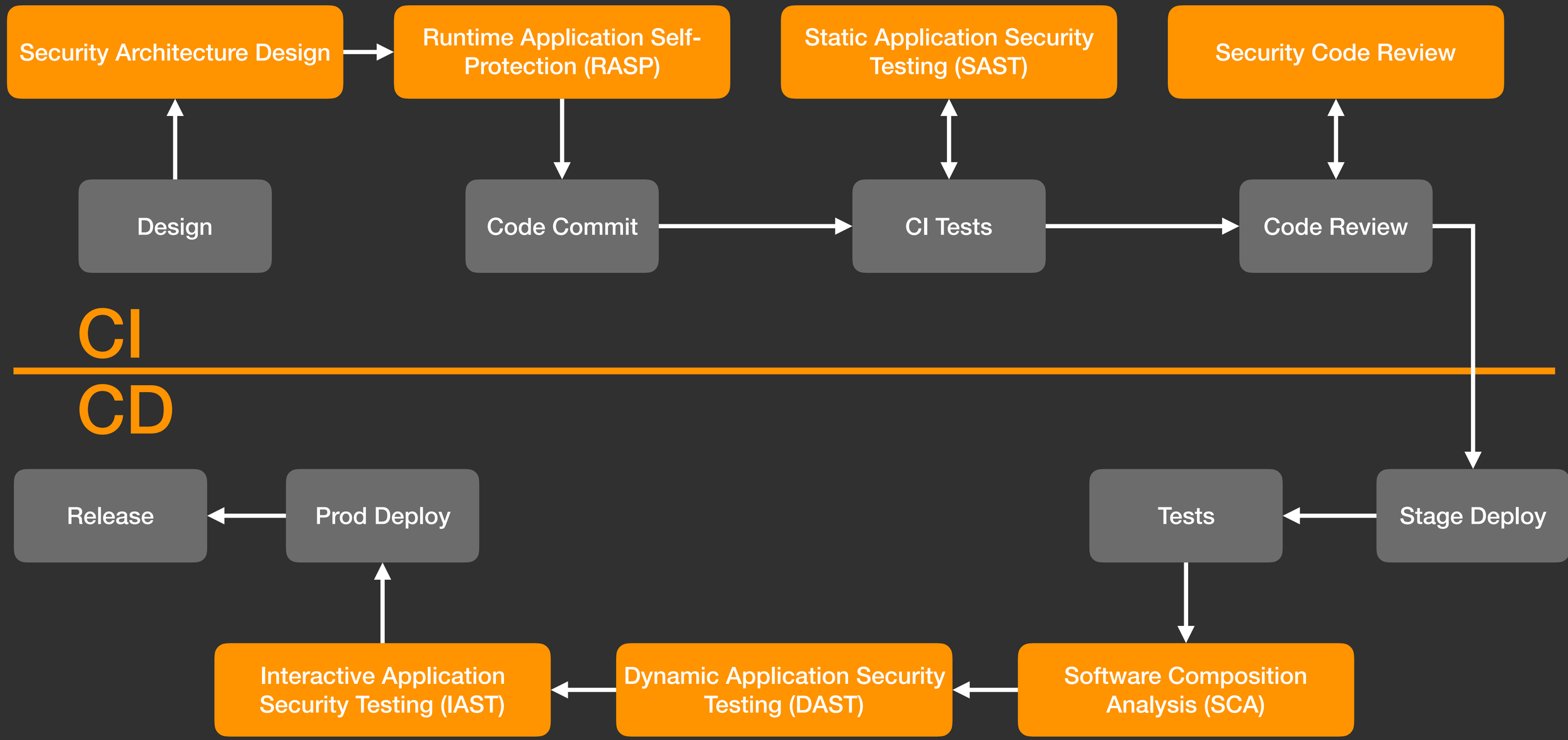
CD 階段、環境互動測試、軟體組建分析（SCA）、動態測試（DAST）、資安配置

資安系統架構、存取控制（Access Control）、機密資訊管理（Secret Management）

CI 階段、靜態掃描（SAST）、程式碼資安檢查（Code Review）、安全環境打包


Retrospective
Daily Scrum
Planning
Review
Product Backlog
TODO List
Increment

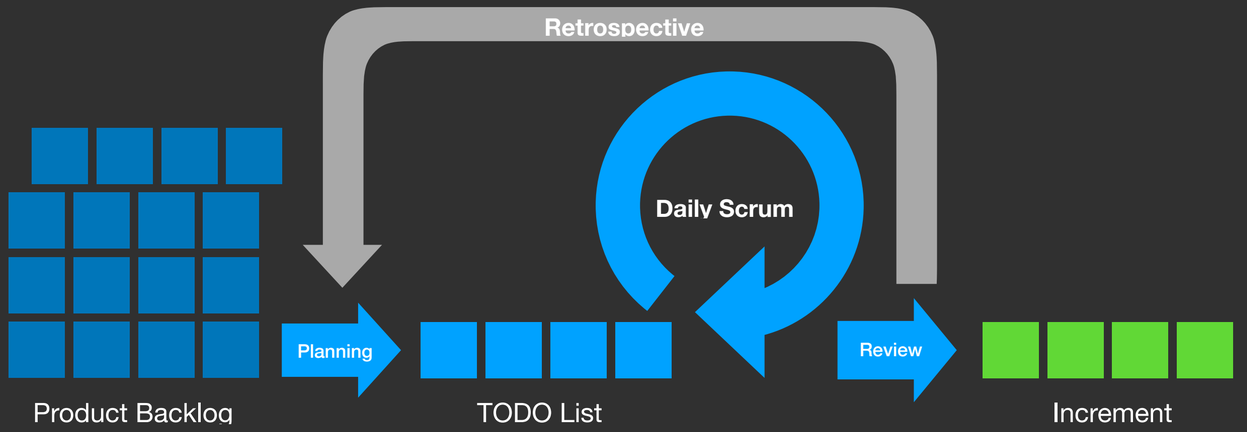

Design
Code Commit
CI Tests
Code Review
CI
CD
Release
Prod Deploy
Tests
Stage Deploy

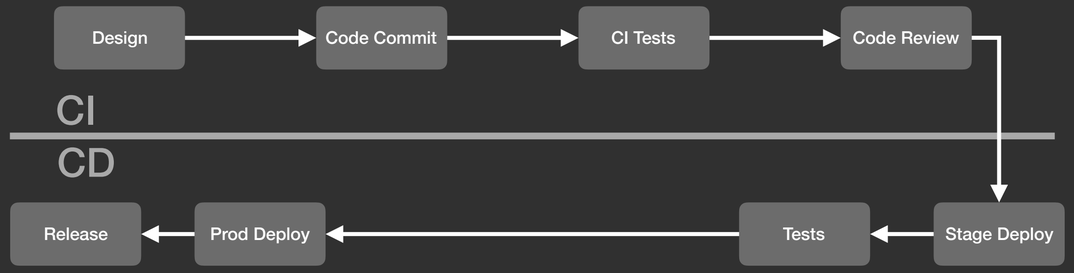


https://secview.io/posts/9-ci-cd/

DevOps & Security

- Waterfall vs DevOps
- Security in DevOps
- CI / CD





Security Testing

自動化

Q & A

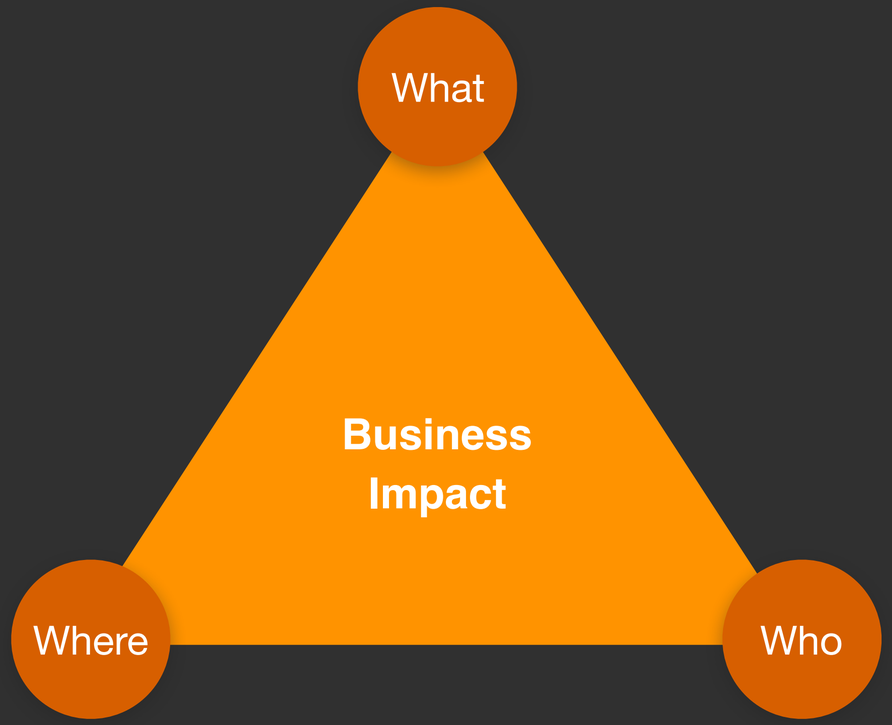
- 商業價值
- 標準化
- 風險評估
- 威脅模型

- 測試範圍
- OWASP Testing Guide V4
- 測試自動化

- 自動化過濾
- 弱點分析、回報
- 資安測試優化

威脅模型

- （What）你要保護什麼？金鑰？客戶資料？
- （Who）誰會攻擊？Script Kiddie？
- （Where）從哪攻擊？網站？功能？人？
- （How）怎麼攻擊？開源工具掃描？社交工程？



資料外洩

資安意識不足

錯誤配置

2022 年時會有 95% 的雲端資安威脅源自人為疏失。

– Gartner
https://www.bnext.com.tw/article/53826/awareness-cybersecurity-cloud

標準化

- 威脅模型（Threat Modeling）
  - STRIDE、PASTA、...
- 資安測試（Security Testing）
  - OWASP Testing Guide V4
  - OWASP ASVS 3.0
- 弱點評估（Vulnerability Assessment）
  - Common Vulnerability Scoring System (CVSS) 3.1

OWASP Testing Guide V4

https://www.owasp.org/index.php/OWASP_Testing_Guide_v4_Table_of_Contents

- Information Gathering
- Configuration and Deploy Management Testing
- Identity Management Testing
- Authentication Testing
- Authorization Testing
- Session Management Testing
- Input Validation Testing
- Error Handling
- Cryptography
- Business Logic Testing
- Client Side Testing

靜態資安測試
(SAST)

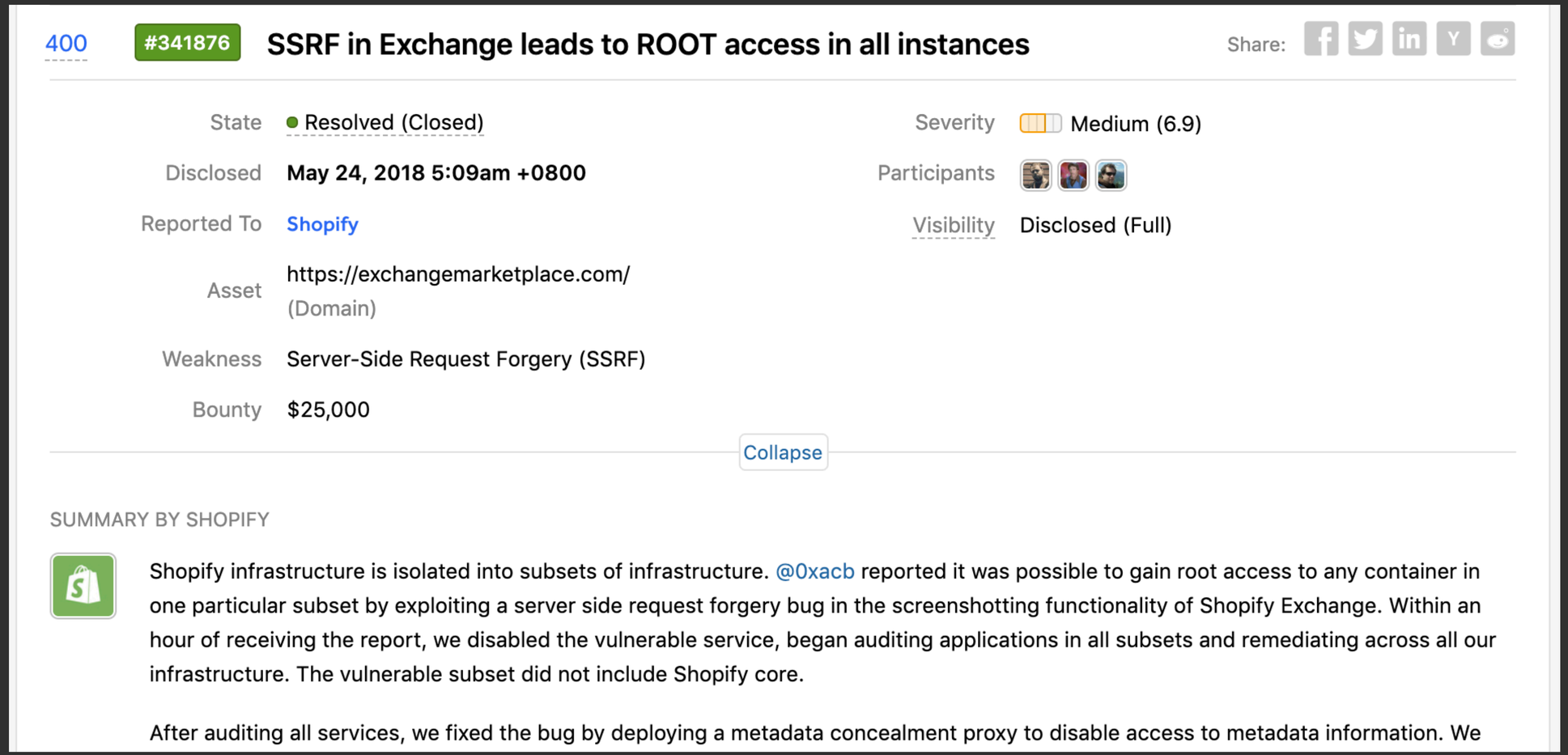
動態資安測試
(DAST)

Web Vulnerabilities ?

AWS S3 Information Leak

https://github.com/nagwww/s3-leaks

Date	Description	
Mar 2018	Medical Records and Patient-Doctor Recordings Were Exposed	information for employe as well as personally id nearly 3,000 individuals unsecured
Mar 2018	Jewelry site accidentally leaks personal details (and plaintext passwords!) of 1.3M users	addresses, zip-codes, addresses. He also clai plaintext passwords
Feb	S3 bucket open to world : Octoly	real names, addresses, addresses
Jan 22	Sensitive medical records on AWS bucket found to be publicly accessible	
Dec 2017	Alteryx leave S3 bucket open for anonymous user : 120m american households exposed	Home addresses, conta status, financial historie
Nov 2017	111 GB of internal customer information from National Credit Federation, a Tampa, Florida-based credit repair service	- SSN - Drivers licesne,
Nov 2017	Uber, the hack happend couple months back was brought to light in Nov 2017>	personal information of driver's license number
Nov 2017	NSA leak exposes Red Disk, the Army's failed intelligence system	100 gigabytes of data f project, codenamed "R
Nov 2017	Australia data leak: Nearly 50,000 government and private staffers' sensitive data publicly exposed	S3 bucket left open by
Oct 2017	How A Cloud Leak Exposed Accenture's Business	



400 #341876 SSRF in Exchange leads to ROOT access in all instances

Share:

State: Resolved (Closed)
Disclosed: May 24, 2018 5:09am +0800
Reported To: Shopify
Asset: https://exchangemarketplace.com/ (Domain)
Weakness: Server-Side Request Forgery (SSRF)
Bounty: $25,000

Severity: Medium (6.9)
Participants:
Visibility: Disclosed (Full)

Collapse

SUMMARY BY SHOPIFY

Shopify infrastructure is isolated into subsets of infrastructure. @0xacb reported it was possible to gain root access to any container in one particular subset by exploiting a server side request forgery bug in the screenshotting functionality of Shopify Exchange. Within an hour of receiving the report, we disabled the vulnerable service, began auditing applications in all subsets and remediating across all our infrastructure. The vulnerable subset did not include Shopify core.

After auditing all services, we fixed the bug by deploying a metadata concealment proxy to disable access to metadata information. We

SSRF to Root Access

https://hackerone.com/reports/341876

The npm Blog

Blog about npm things.

Reported malicious module: getcookies

Early May 2nd, the npm security team received and responded to reports of a package that masqueraded as a cookie parsing library but contained a malicious backdoor. The result of the investigation concluded with three packages and three versions of a fourth package being unpublished from the npm Registry.

No packages published to the npm Registry used the malicious modules in a way that would have allowed the backdoor to be triggered. Applications not published to the registry that directly required the malicious modules might have been vulnerable, but are out of the scope of our analysis.

Initial report

Initial information from the community reported that the package `getcookies` contained a potential backdoor, that `express-cookies` and `http-fetch-cookies` depended upon getcookies, and that a popular package, `mailparser`, depended upon http-fetch-cookies.

Triage

Upon receiving the report, npm's security team started triage. The goal of triage was determining

- Information Gathering
- Configuration and Deploy Management Testing
- Identity Management Testing
- Authentication Testing
- Authorization Testing
- Session Management Testing
- Data Validation Testing
- Error Handling
- Cryptography
- Business Logic Testing
- Client Side Testing

- Information Gathering
- Configuration and Deploy Management Testing
- Identity Management Testing
- Authentication Testing
- Authorization Testing
- Session Management Testing
- Data Validation Testing
- Error Handling
- Cryptography
- Business Logic Testing
- Client Side Testing

- Shodan: https://www.shodan.io/
- Censys: https://censys.io/
- Sublist3r: https://github.com/aboul3la/Sublist3r
- Nmap: https://nmap.org/

- Information Gathering
- Configuration and Deploy Management Testing
- Identity Management Testing
- Authentication Testing
- Authorization Testing
- Session Management Testing
- Data Validation Testing
- Error Handling
- Cryptography
- Business Logic Testing
- Client Side Testing

- Kube Hunter: https://github.com/aquasecurity/kube-hunter
- Nginx configuration: https://github.com/yandex/gixy
- SSLScan: https://github.com/rbsec/sslscan

- Information Gathering
- Configuration and Deploy Management Testing
- Identity Management Testing
- Authentication Testing
- Authorization Testing
- Session Management Testing
- Data Validation Testing
- Error Handling
- Cryptography
- Business Logic Testing
- Client Side Testing

- SAST Tools
- Python: https://github.com/PyCQA/bandit
- Go: https://github.com/securego/gosec
- ...

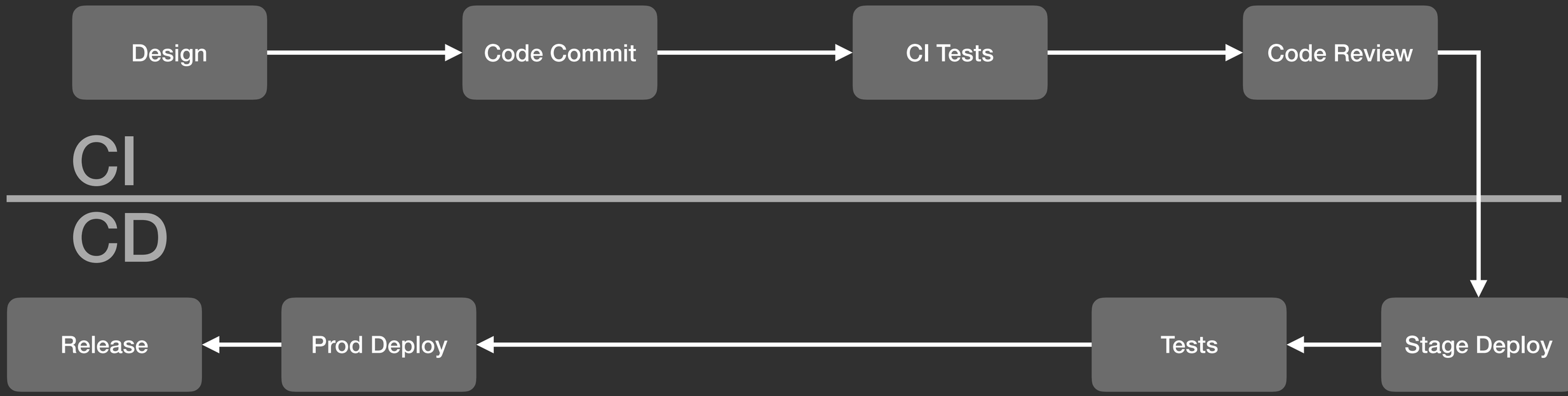
- Information Gathering
- Configuration and Deploy Management Testing
- Identity Management Testing
- Authentication Testing
- Authorization Testing
- Session Management Testing
- Data Validation Testing
- Error Handling
- Cryptography
- Business Logic Testing
- Client Side Testing

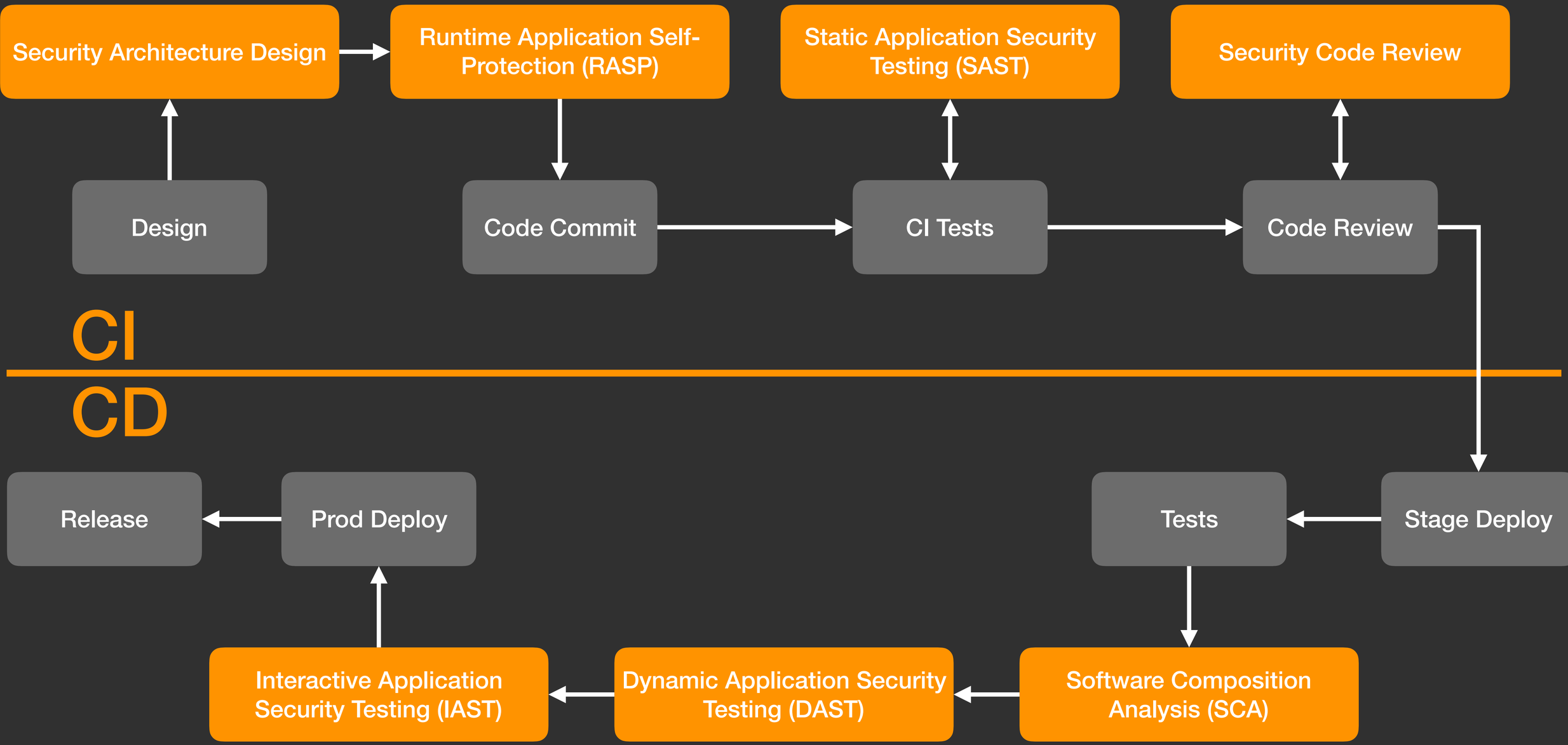
- DAST Tools
- OWASP ZAP: https://github.com/zaproxy/zaproxy/wiki/ZAP-API-Scan
- Nitkto2: https://github.com/sullo/nikto
- Sqlmap: https://github.com/sqlmapproject/sqlmap
- Arachni: https://github.com/Arachni/arachni
- Behave: https://github.com/behave/behave

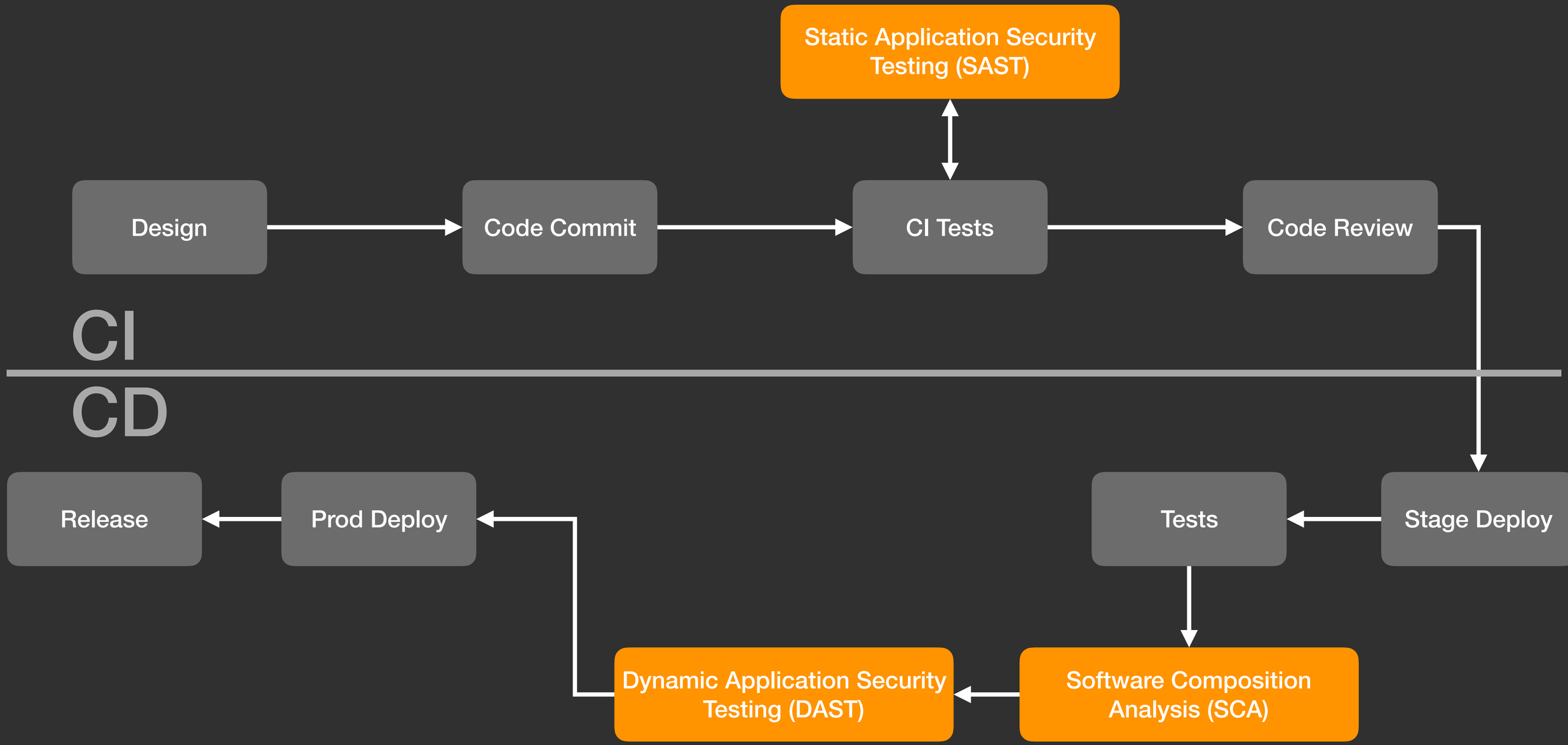
- Information Gathering
- Configuration and Deploy Management Testing
- Identity Management Testing
- Authentication Testing
- Authorization Testing
- Session Management Testing
- Data Validation Testing
- Error Handling
- Cryptography
- Business Logic Testing
- Client Side Testing

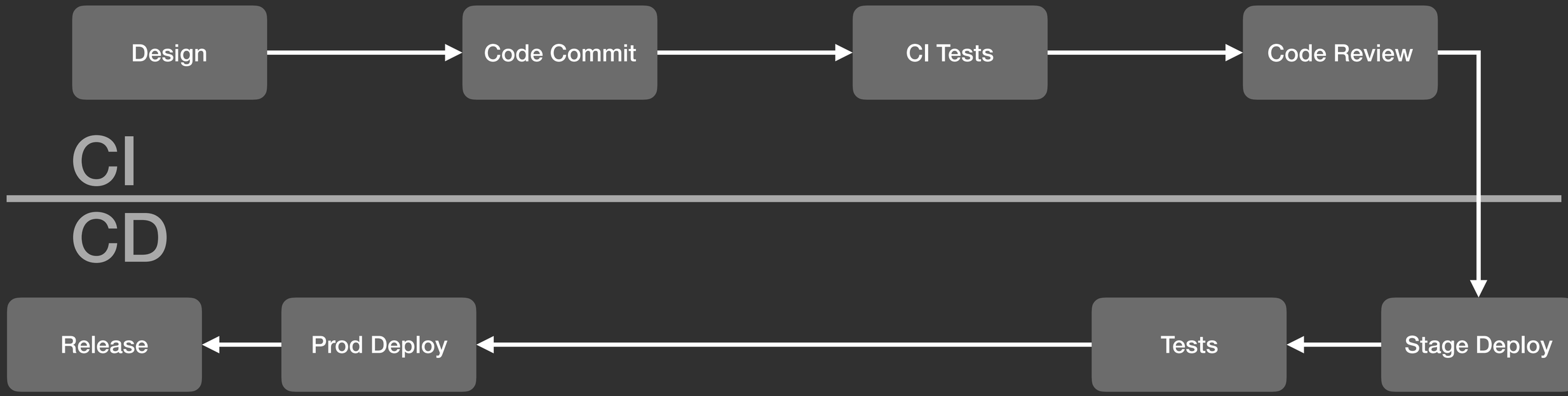
- Software Composition Analysis (SCA)
- JS libraries: https://retirejs.github.io/retire.js/
- 3rd party libraries: https://snyk.io/
- Container analysis: https://github.com/coreos/clair
- Vuls: https://github.com/future-architect/vuls

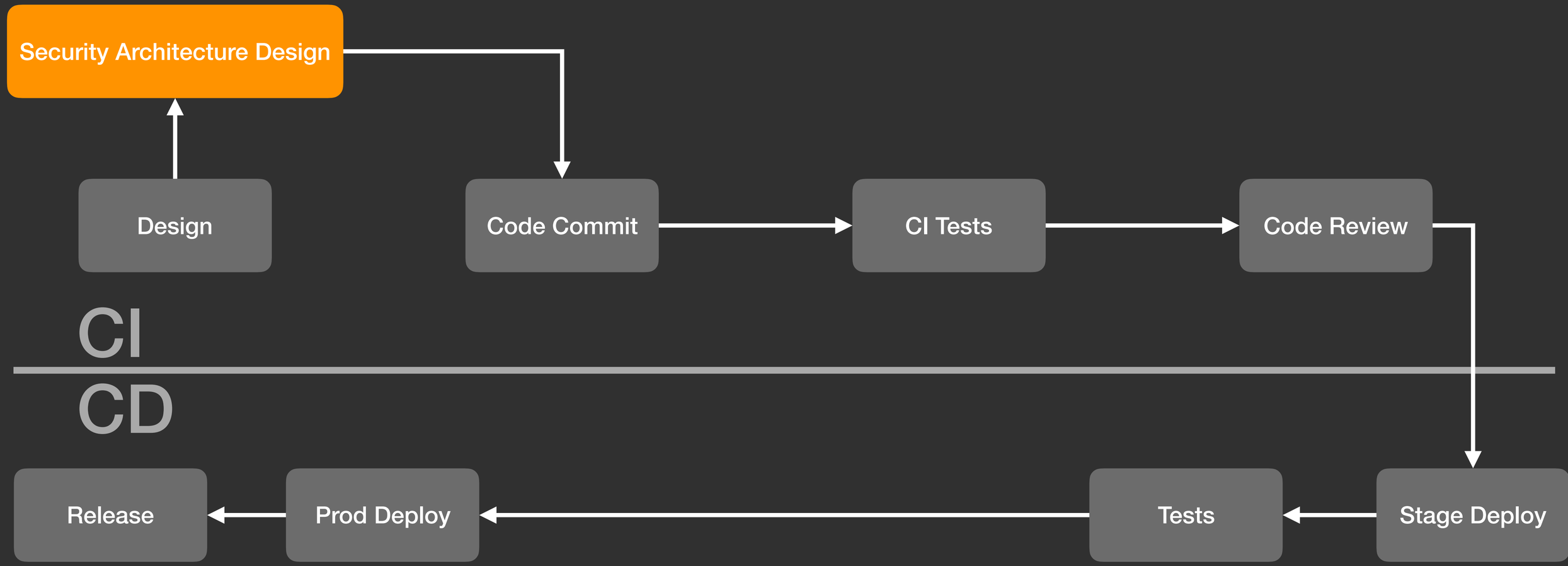
自動化

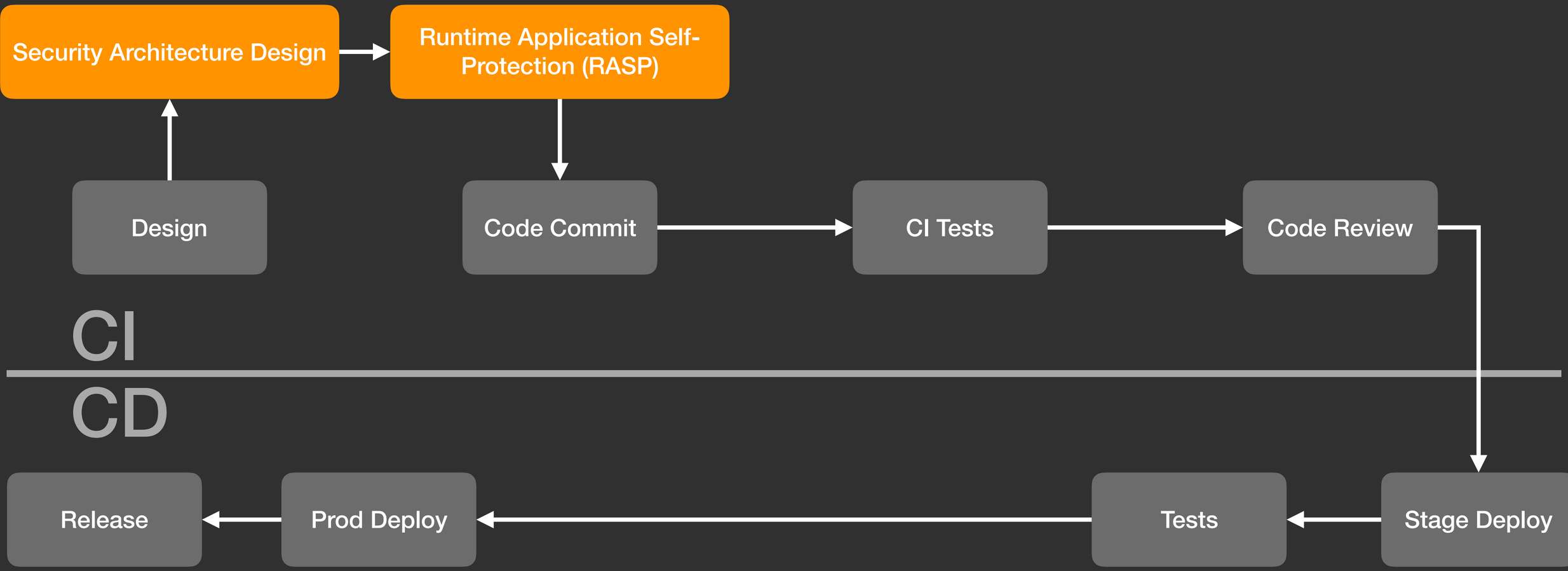

Design
Code Commit
CI Tests
Code Review
CI
CD
Release
Prod Deploy
Tests
Stage Deploy

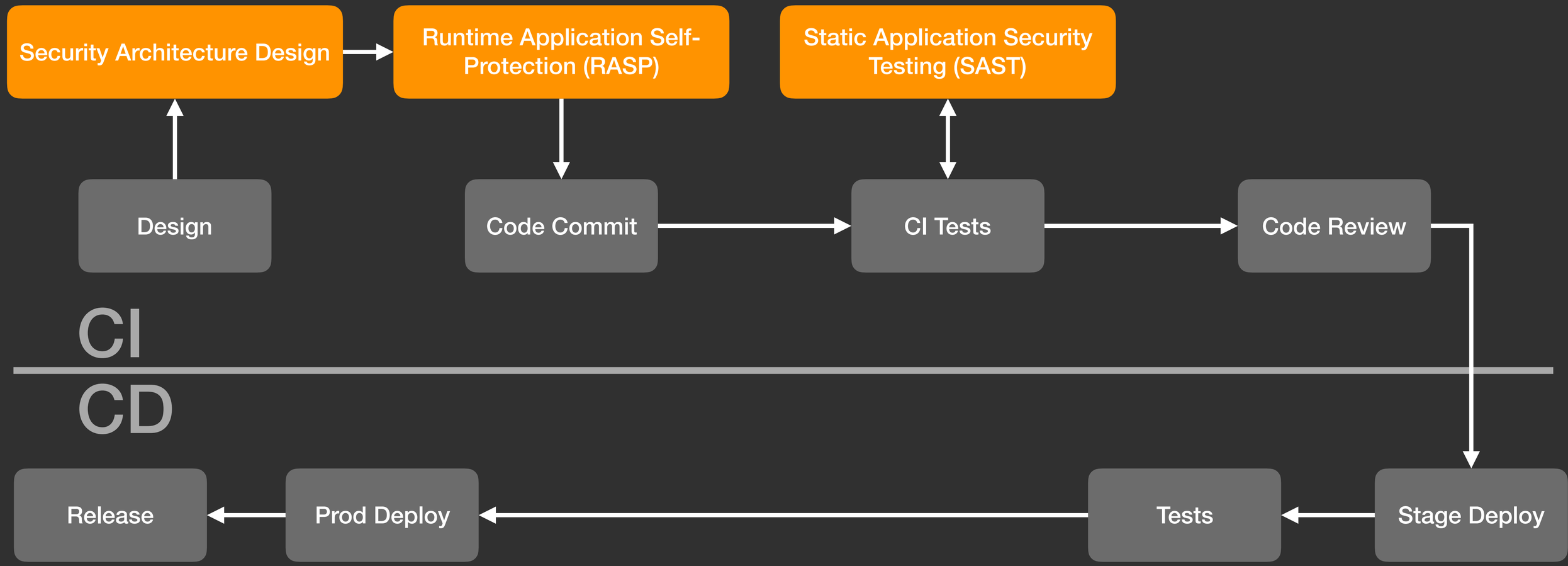

Security Architecture Design
Runtime Application Self-Protection (RASP)
Static Application Security Testing (SAST)
Security Code Review
Design
Code Commit
CI Tests
Code Review
CI
CD
Release
Prod Deploy
Tests
Stage Deploy
Interactive Application Security Testing (IAST)
Dynamic Application Security Testing (DAST)
Software Composition Analysis (SCA)

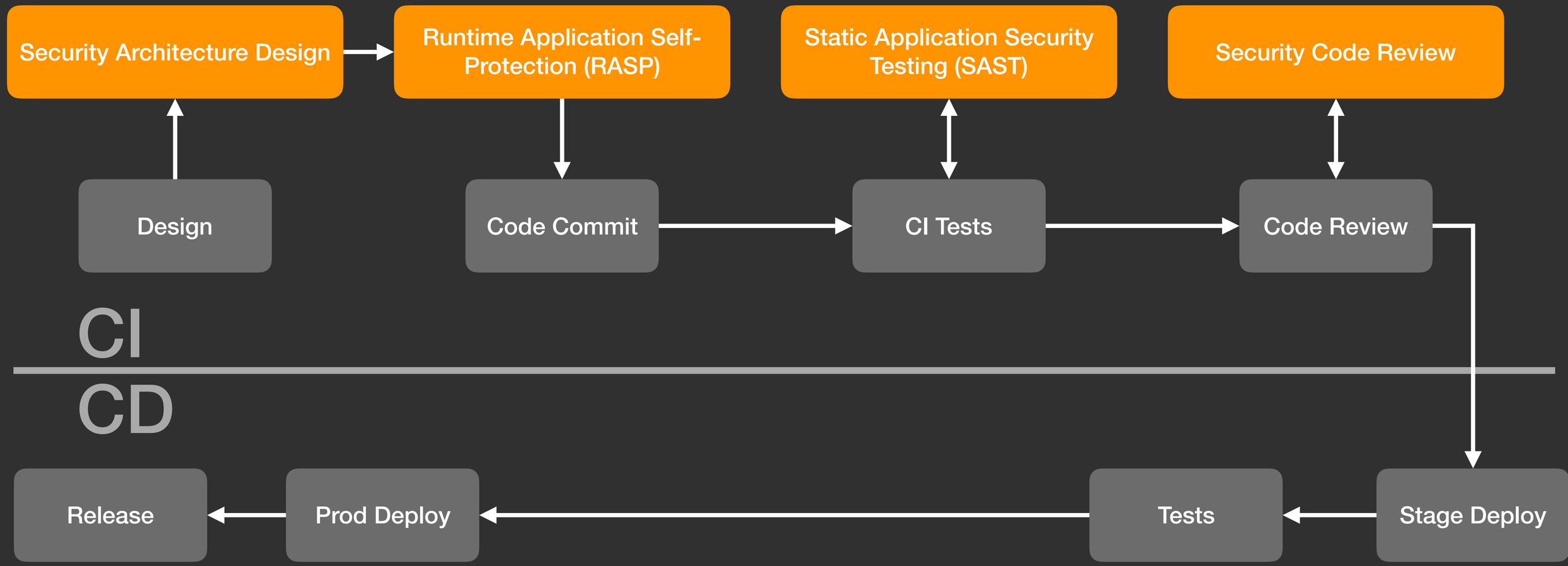

Static Application Security Testing (SAST)
Design
Code Commit
CI Tests
Code Review
CI
CD
Release
Prod Deploy
Tests
Stage Deploy
Dynamic Application Security Testing (DAST)
Software Composition Analysis (SCA)

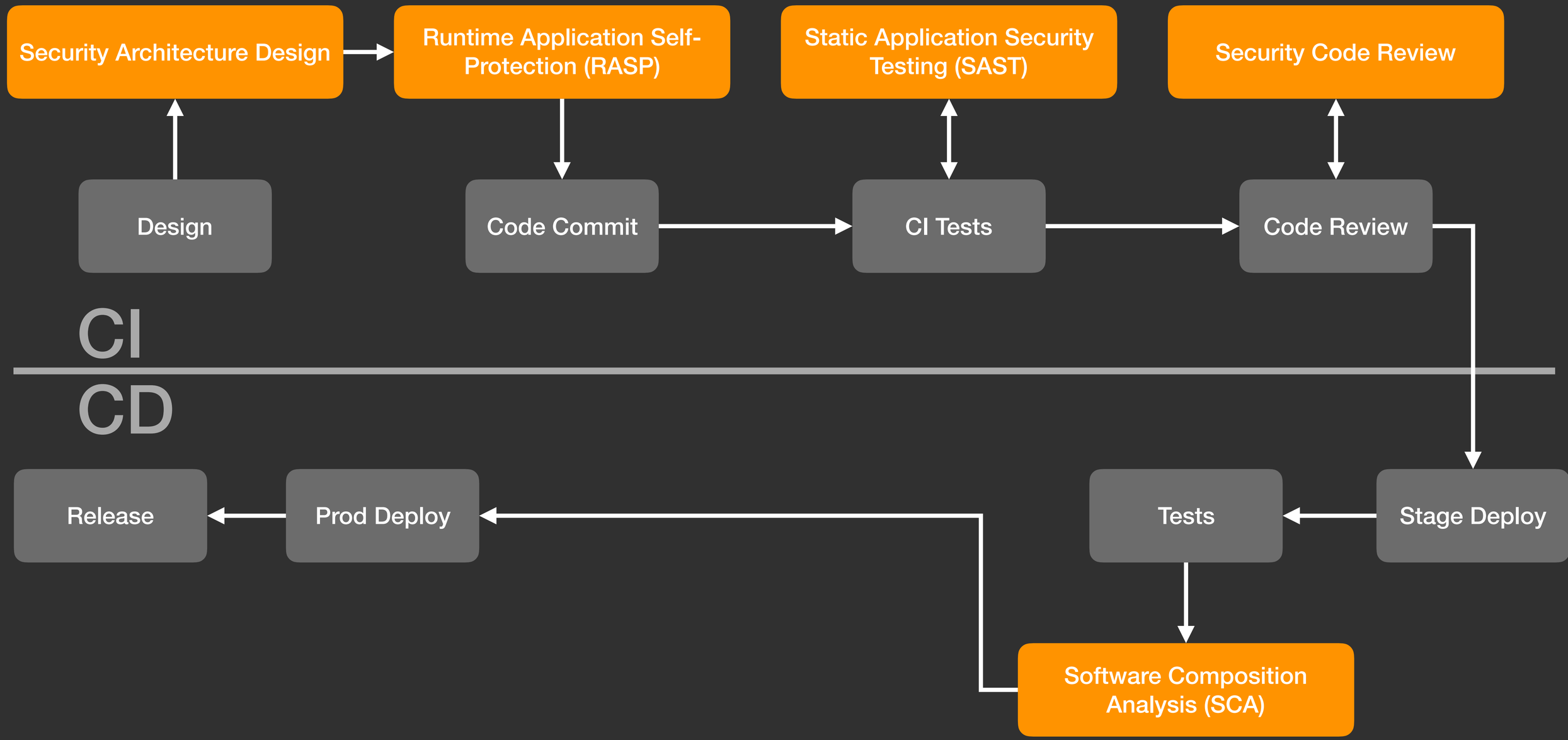

Design
Code Commit
CI Tests
Code Review
CI
CD
Release
Prod Deploy
Tests
Stage Deploy

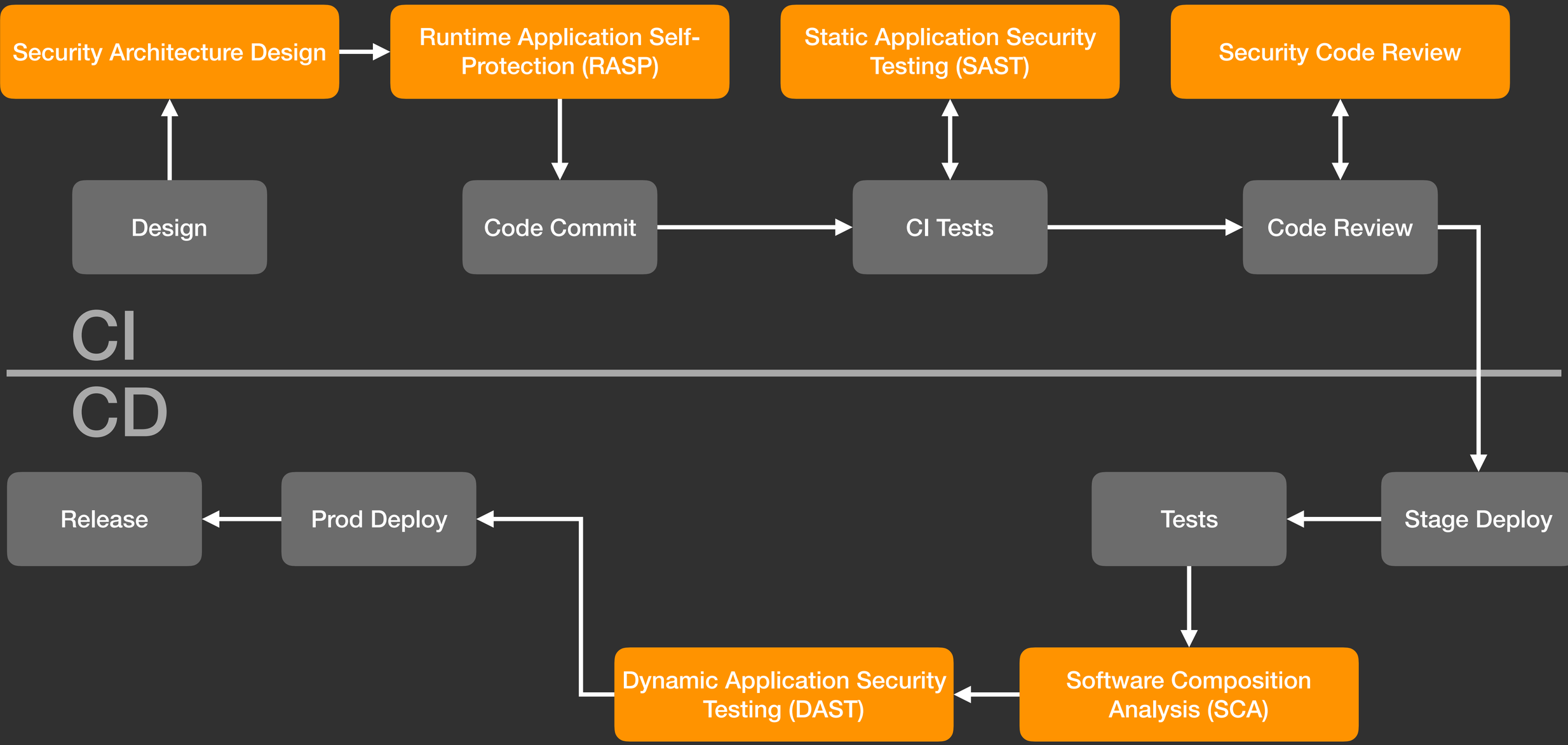

Security Architecture Design
Design
Code Commit
CI Tests
Code Review
CI
CD
Release
Prod Deploy
Tests
Stage Deploy

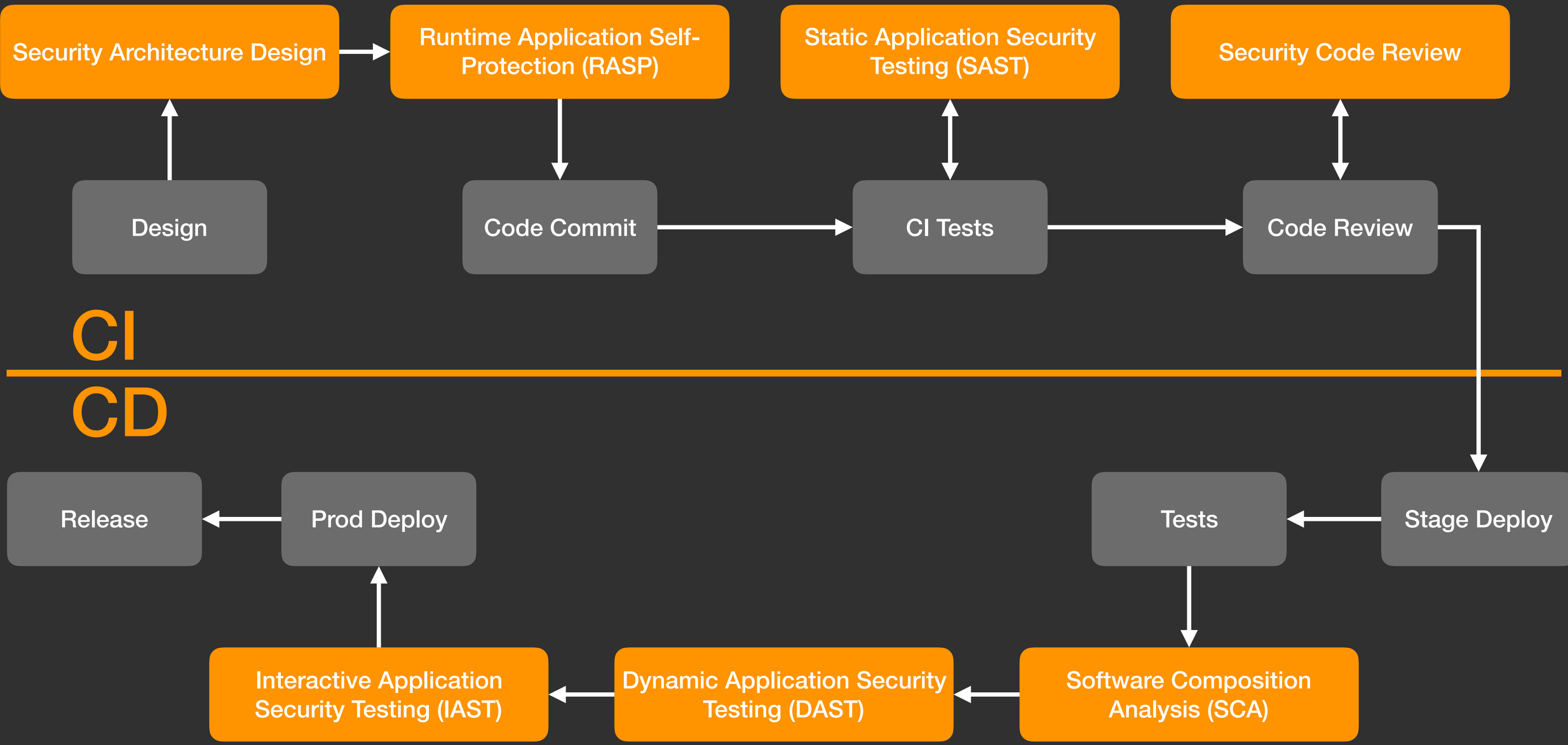

Security Architecture Design
Runtime Application Self-Protection (RASP)
Design
Code Commit
CI Tests
Code Review
CI
CD
Release
Prod Deploy
Tests
Stage Deploy


Security Architecture Design
Runtime Application Self-Protection (RASP)
Static Application Security Testing (SAST)
Design
Code Commit
CI Tests
Code Review
CI
CD
Release
Prod Deploy
Tests
Stage Deploy

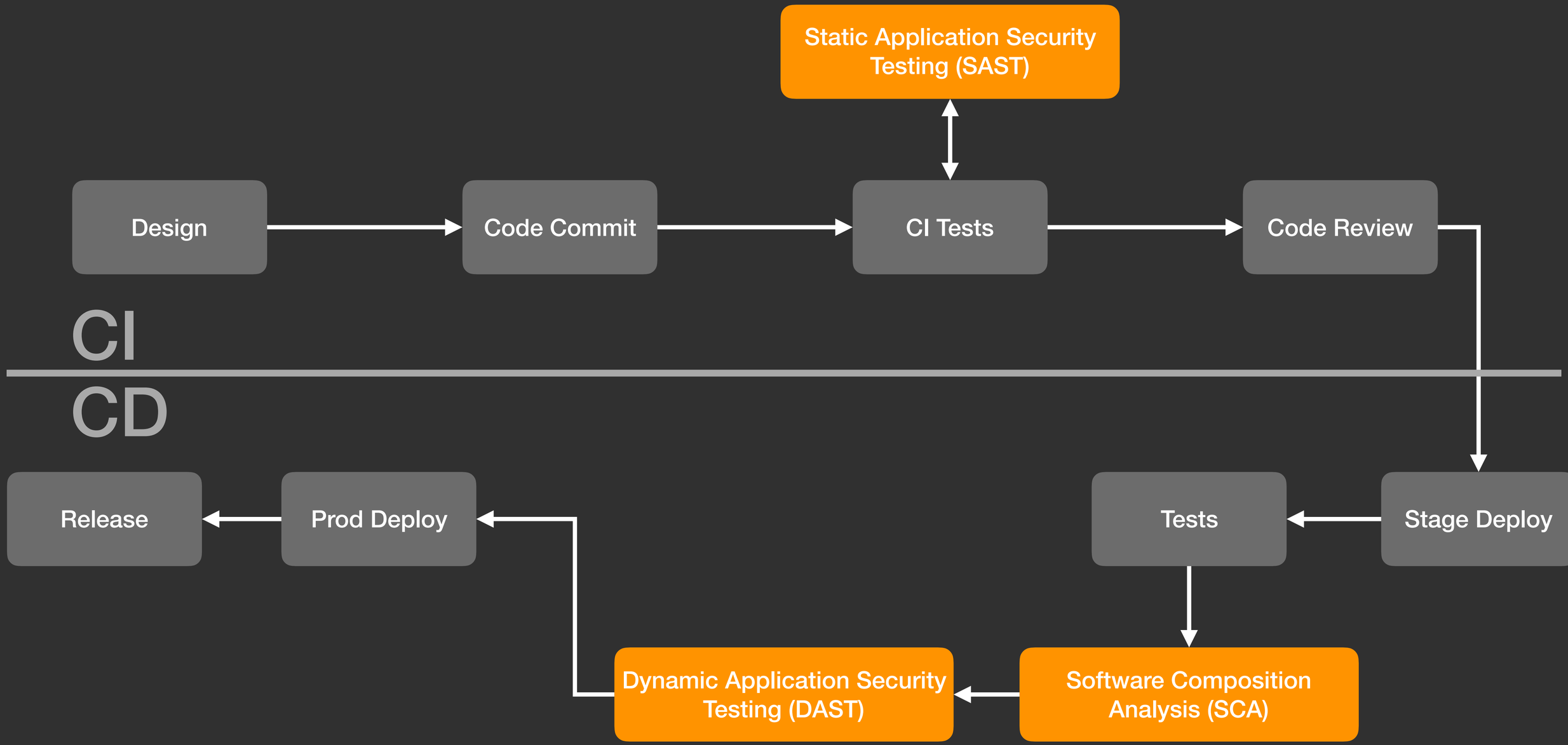

Security Architecture Design
Runtime Application Self-Protection (RASP)
Static Application Security Testing (SAST)
Security Code Review
Design
Code Commit
CI Tests
Code Review
CI
CD
Release
Prod Deploy
Tests
Stage Deploy


Security Architecture Design
Runtime Application Self-Protection (RASP)
Static Application Security Testing (SAST)
Security Code Review
Design
Code Commit
CI Tests
Code Review
CI
CD
Release
Prod Deploy
Tests
Stage Deploy
Software Composition Analysis (SCA)


Security Architecture Design
Runtime Application Self-Protection (RASP)
Static Application Security Testing (SAST)
Security Code Review
Design
Code Commit
CI Tests
Code Review
CI
CD
Release
Prod Deploy
Tests
Stage Deploy
Dynamic Application Security Testing (DAST)
Software Composition Analysis (SCA)


Security Architecture Design
Runtime Application Self-Protection (RASP)
Static Application Security Testing (SAST)
Security Code Review
Design
Code Commit
CI Tests
Code Review
CI
CD
Release
Prod Deploy
Tests
Stage Deploy
Interactive Application Security Testing (IAST)
Dynamic Application Security Testing (DAST)
Software Composition Analysis (SCA)


預防
自動化測試
滲透測試
漏洞獎勵計劃
未找到
利用


Static Application Security Testing (SAST)
Design
Code Commit
CI Tests
Code Review
CI
CD
Release
Prod Deploy
Tests
Stage Deploy
Dynamic Application Security Testing (DAST)
Software Composition Analysis (SCA)

Static Application Security Testing (SAST)

靜態資安測試（SAST）

- Security linter、錯誤配置檢測
- 需要低偽陽性（False Positive）
- 快速檢查

動態資安測試（DAST）

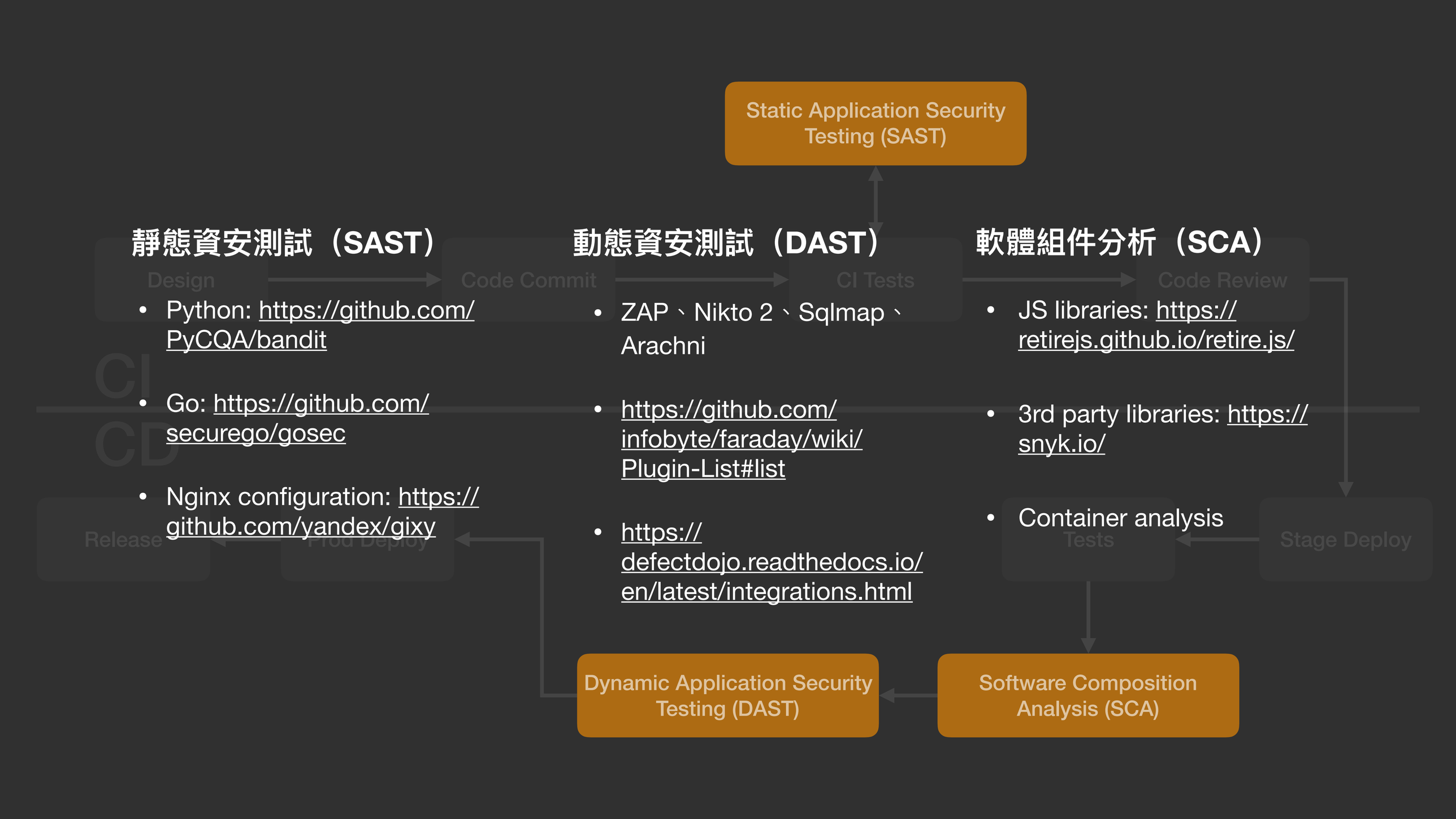
- 輸入驗證測試
- 資訊收集、部署、認證授權測試
- 降低報告的內容和數量

軟體組件分析（SCA）

- 相依性套件
- 第三方函式庫、開發框架、Docker 映像檔、系統軟體套件

Dynamic Application Security Testing (DAST)

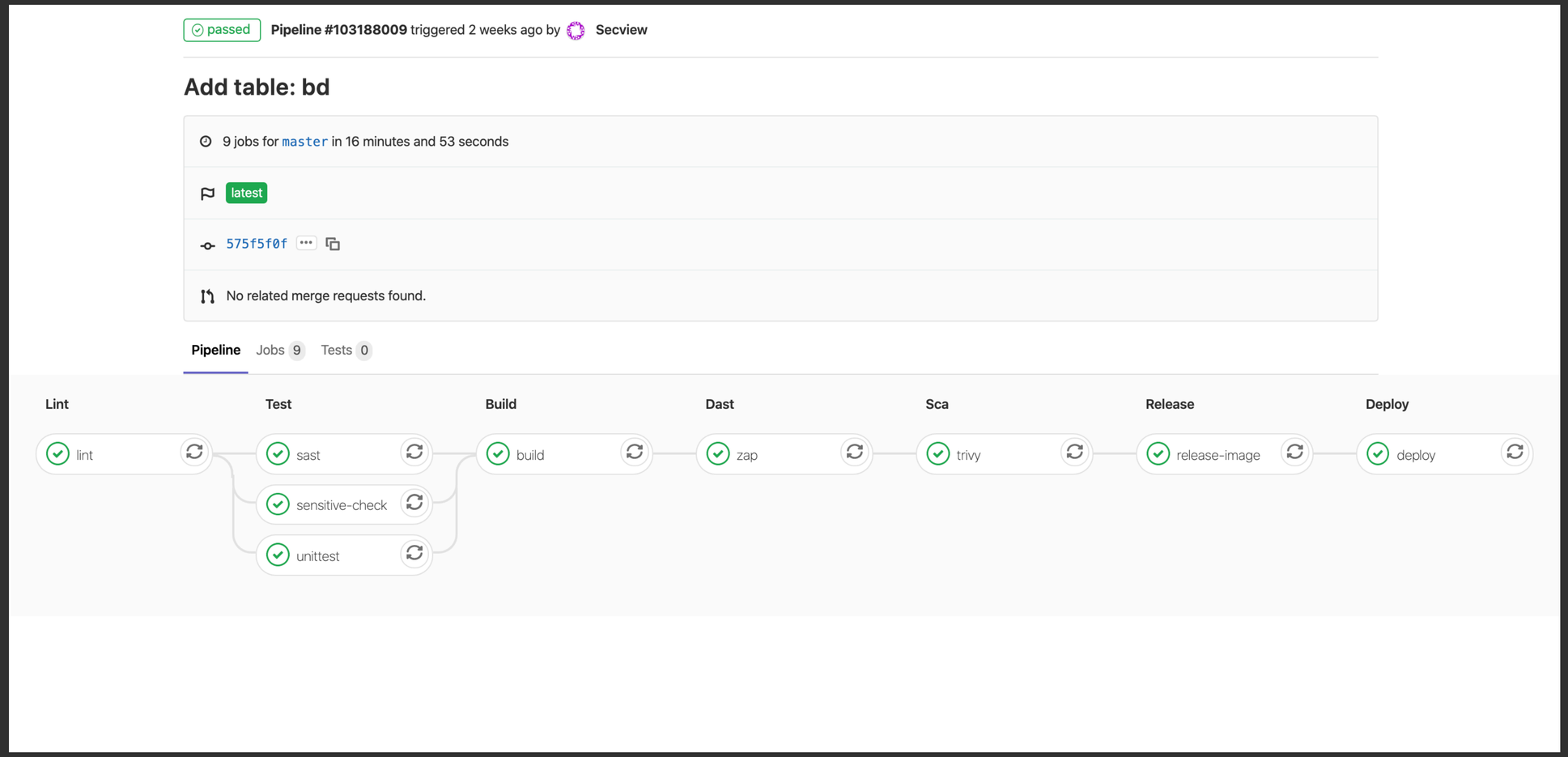
Software Composition Analysis (SCA)



- Python: https://github.com/PyCQA/bandit
- Go: https://github.com/securego/gosec
- Nginx configuration: https://github.com/yandex/gixy

- ZAP、Nikto 2、Sqlmap、Arachni
- https://github.com/infobyte/faraday/wiki/Plugin-List#list
- https://defectdojo.readthedocs.io/en/latest/integrations.html

- JS libraries: https://retirejs.github.io/retire.js/
- 3rd party libraries: https://snyk.io/
- Container analysis

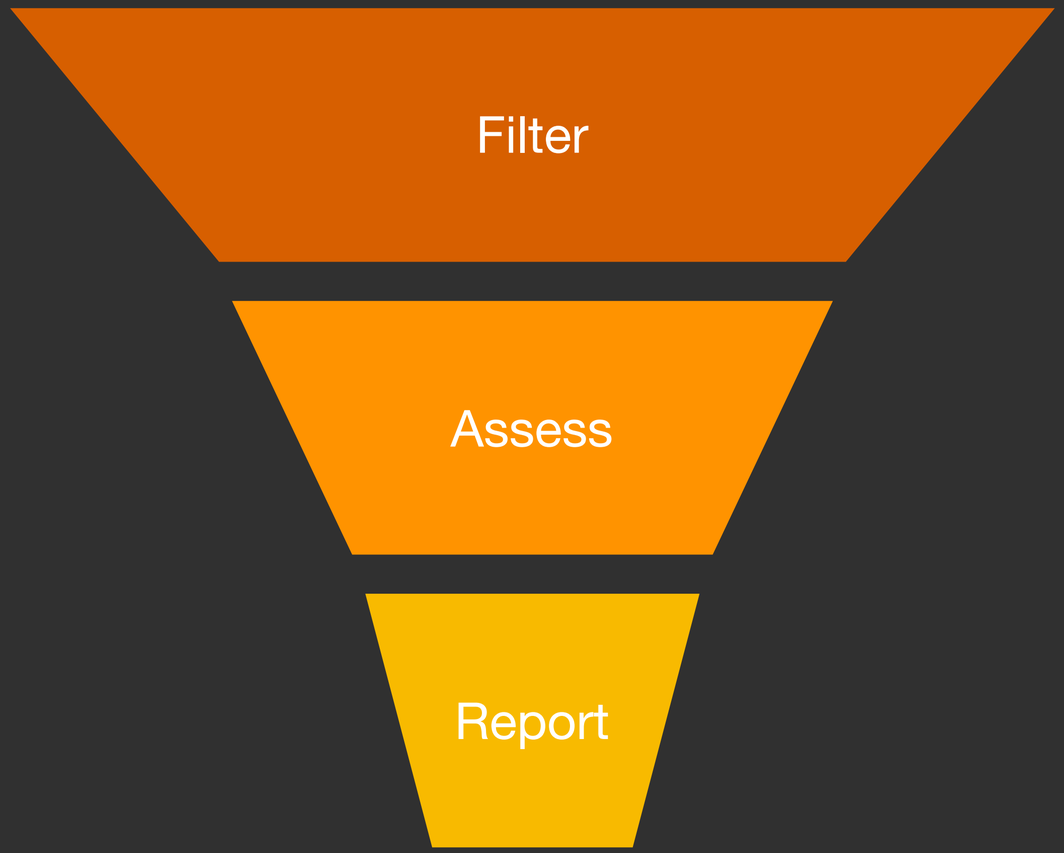


Security Testing in CI / CD

SAST: https://secview.io/posts/13-hitcon-defense-summit-2019/

漏洞管理

- 自動化過濾（Filter）
  - 偽陽性、重複掃到的問題（Duplicate Issues）
- 嚴重性評估（Assess）
  - CVSS 3.1
- 報告（Report）
  - 嚴重性、優先順序、漏洞簡介、漏洞影響
  - 建議修復方法
- DefectDojo: https://github.com/DefectDojo/django-DefectDojo



- 商業價值
- 標準化
- 風險評估
- 威脅模型

- 測試範圍
- OWASP Testing Guide V4
- 測試自動化

- 自動化過濾
- 弱點分析、回報
- 資安測試優化

標準化

資安測試項目、風險量化、威脅模型分析

模組化

Security-as-code
資安程式化、資安架構、重複使用

自動化

測試、監控、過濾、回報與反應自動化、持續疊代

Shift Left

及早發現

持續檢測

降低成本


SECURITY SPECIALIST


INCIDENT RESPONDER


SECURITY
ADMINISTRATOR


VULNERABILITY
ASSESSOR


CRYPTOGRAPHER


SECURITY MANAGER


SECURITY ARCHITECT

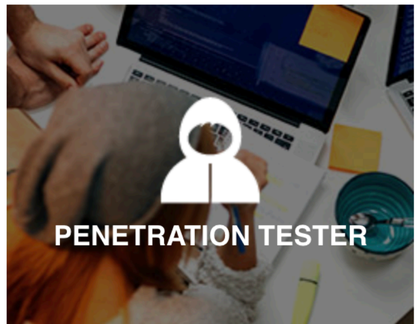

CHIEF INFORMATION
SECURITY OFFICER


SECURITY ANALYST


SECURITY AUDITOR

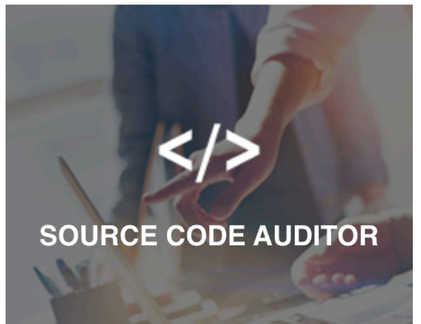

SECURITY DIRECTOR


FORENSIC EXPERT


PENETRATION TESTER


SECURITY CONSULTANT


SECURITY ENGINEER


SOURCE CODE AUDITOR

KEEP DATA SAFE

Protect your company's data while you learn new skills and assess preventive security measures with these entry-level cyber security careers.

- Security Engineer
- Security Analyst
- Incident Responder
- Security Specialist

STAY ONE STEP AHEAD

Thwart cybercrime by thinking like a hacker as you test and implement layered security systems and keep your company's data safe from cybercriminals.

- Vulnerability Assessor
- Penetration Tester
- Source Code Auditor
- Security Auditor

FIND YOUR SPECIALTY

Fine tune your expertise by focusing on these specialized cyber security career fields. Become an expert in your chosen area of cybercrime prevention.

- Forensics Expert
- Security Consultant
- Cryptographer
- Security Engineer

LEAD YOUR TEAM

Reach the pinnacle of the profession as a leader, and satisfy business goals while making decisions about your company's security and integrity.

- Security Administrator
- Cyber Security Manager
- Security Director
- Security Architect
- Chief Information Security Officer (CISO)

Q & A