

INTRODUCTION TO DEVOPS RED TEAMING





ABOUT CYBERWARFARE LABS:

CW Labs is a Global Infosec company specializing in cybersecurity practical learning situated across UK, US & India. The company has 2 primary divisions:

- 1. Niche Cyber Range Labs
- 2. Continuous Learning: Infinity Platform



INFINITE LEARNING EXPERIENCE



ABOUT SPEAKER

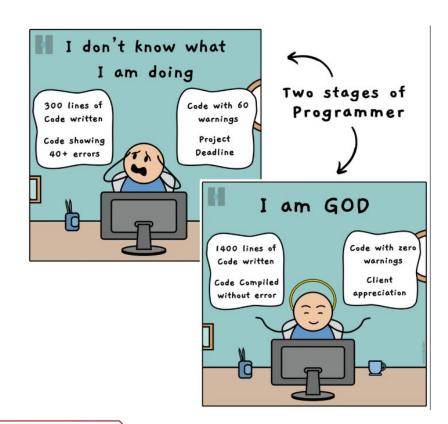
<u>Abhijeet Kumar</u> (Security Researcher)

His research areas include Red Team Operations, Network Security, Cloud Infrastructure, and Linux Systems. Apart from this, he enjoys researching Adversarial TTPs and experimenting in his homelab.



SOFTWARE DEVELOPMENT 101

- ★ A set of recurring activities involved in the lifecycle of a software product
- ★ These activities include :-
 - Analysis / Plan
 - Design
 - Development
 - Test
 - Release / Deployment
 - Maintenance





SOFTWARE DEVELOPMENT LIFE CYCLE



Source: Zitoc

project is built. Developers use programming languages

for the assigned project and complete the project.

project along with its communication and

data flow representation.



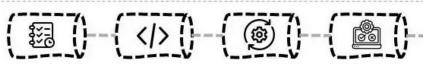
DEVOPS 101

- ★ DevOps is a software development methodology that streamlines the development cycle.
- ★ It aims to combine development (Dev) and operations (Ops) tasks into a unified discipline.
- ★ DevOps includes tools and practices that enable:
 - Continuous Integration (CI)
 - Continuous Delivery (CD)
 - Automation
 - Collaboration



PHASES IN DEVOPS

Dev Ops



PLAN

Requirements

- Workflow

Planning - Task lists

- Sprints

Code

DEVELOP

Shared source code repository

Version controls BUILD

Continuous Integration

- Frror Detection

 Automated tests

TEST

UAT

Performance

Load Testing

Continuous Testing

RELEASE

Repository

- Schedule plan

- Microservices

DEPLOY

Blue-green Strategy

- Configuration

 Automated deployment

Multi-level

OPERATE MONITOR

Environment Infrastructure

Notifications

Recovery Logging

Feedback

- Data Collection

- Productivity

Continuous Integration

Continuous Delivery

Continuous Deployment

Continuous Testing

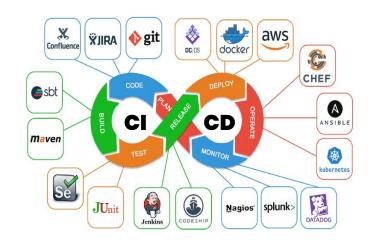
Continuous Operations

Source: Polestarlip



CONTINUOUS INTEGRATION AND CONTINUOUS DELIVERY (CI/CD)

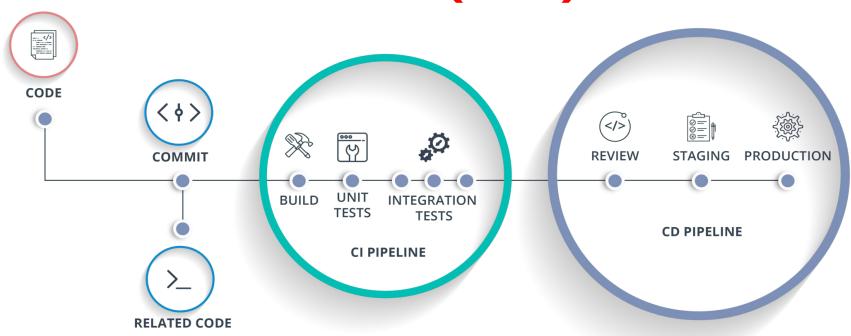
- ★ It's an essential phase within the DevOps framework which automates the code integration and delivery process.
- ★ Bridges the gap between development and operations through automation.



Source: Dev Genius



CONTINUOUS INTEGRATION AND CONTINUOUS DELIVERY (CI/CD)

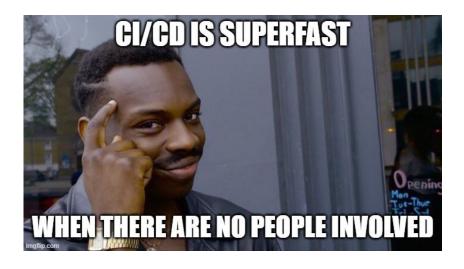


Source: <u>Browserstack</u>



CONTINUOUS INTEGRATION AND CONTINUOUS DELIVERY (CI/CD)

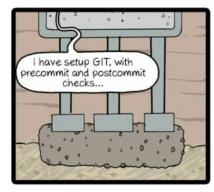
- ★ Commonly used CI/CD platforms include :
 - o AWS Codepipeline
 - Azure DevOps
 - GCP DevOps
 - Jenkins
 - Circle CI
 - Travis CI

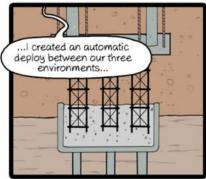


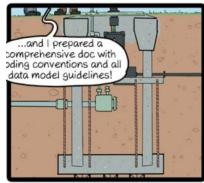


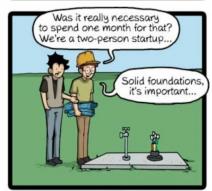
COMPONENTS OF INTEREST

- ★ Components of interest include :-
 - Version Control System
 - GIT
 - Build Tools
 - Compilers
 - Artifacts
 - Artifact Storage
 - Cloud Storage
 - Deploy Tools
 - Virtual Machines
 - Containers







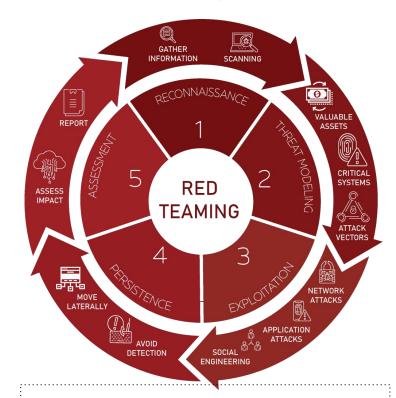


CommitStrip.com



NEED FOR DEVOPS RED TEAMING

- More industry focus on DevSecOps and few organized study materials for Hacking DevOps.
 - With the exception of threat intel reports of APT groups compromising CI/CD pipelines.
 - Some interesting talks by security researchers.
 - Independent blogs



Source: <u>Trolleye Security</u>



DEVOPS THREAT MODEL

Initial access	Execution	Persistence	Privilege escalation	Credential access	Lateral movement	Defense evasion	Impact	Exfiltration
SCM authentication	Poisoned pipeline execution (3)	Change code/pipeline configuration in repository (3)	Secrets stored in private repositories	User credentials	Compromise build artifacts	Service logs manipulation	DDoS using pipeline compute resources	Clone for private repositories
CI/CD service authentication	Dependencies tampering (3)	Inject in artifacts	Commit from pipeline to protected branches	Service credentials	Registry injection	Compilation manipulation (2)	Crypto mining over pipeline compute resources	Access to pipelines logs
Configured webhooks	DevOps resources compromise	Modify images in registry	Certificates and identities from metadata services		Spread from pipeline into deployment resources	Reconfigure branch protections	Local DoS to CI/CD pipelines	Exfiltrate data from production resources
Organization's public repositories	Control of common registry	Create service credentials					Resource deletion	
Endpoint compromise			•			,		•

Source: <u>Microsoft DevOps Threat Matrix</u>



CI/CD SECURITY RISKS

Top 10
CI/CD
Security
Risks



CICD-SEC-2 Inadequate Identity and Access Management

CICD-SEC-3 Dependency Chain Abuse

CICD-SEC-4 Poisoned Pipeline Execution (PPE)

CICD-SEC-5 Insufficient PBAC (Pipeline-Based Access Controls)

CICD-SEC-6 Insufficient Credential Hygiene

CICD-SEC-7 Insecure System Configuration

CICD-SEC-8 Ungoverned Usage of 3rd Party Services

CICD-SEC-9 Improper Artifact Integrity Validation

CICD-SEC-10 Insufficient Logging and Visibility

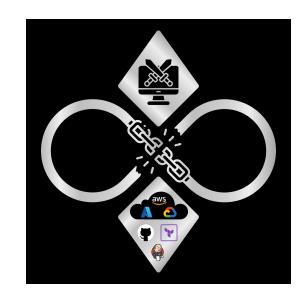


Source: OWASP Top 10 CI/CD Security Risks



DEVOPS RED TEAM ANALYST

- ★ DevOps Red Team Analyst (DO-RTA) course focuses on offensive operations across cloud-native & on-premises CI/CD platforms.
- ★ Course is designed across multiple DevOps pipelines with realistic attack scenarios.



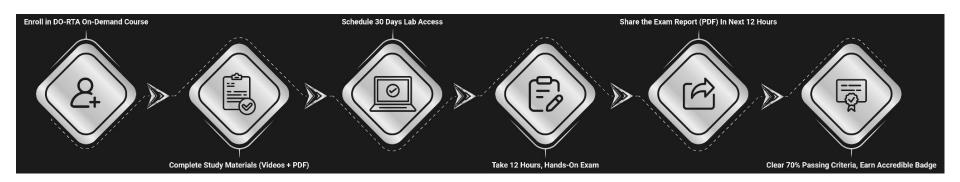


WHAT THIS COURSE IS ABOUT?

- Introduction to DevOps & DevSecOps fundamentals and :-
 - Different phases that comprise them.
 - Services & Tools used during different phases.
 - What the red team operators / penetration testers can do with the initial access with push permissions to Version Control Service (VCS).
- The course will focus on exploiting default configurations & common misconfigurations that occur during DevOps lifecycle.

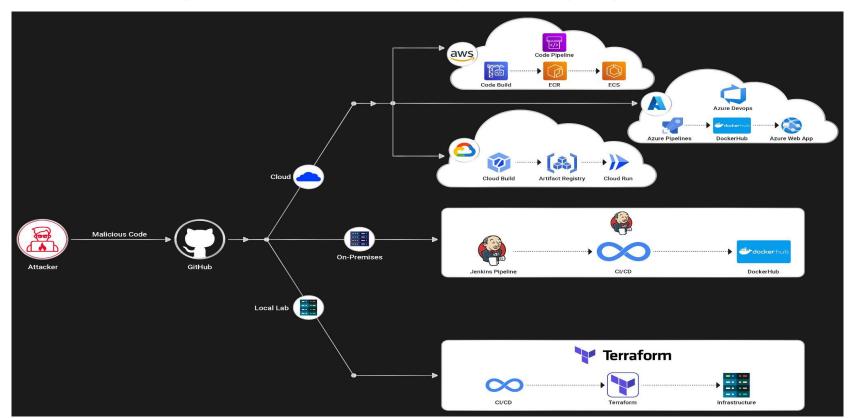


DO-RTA COURSE FLOW

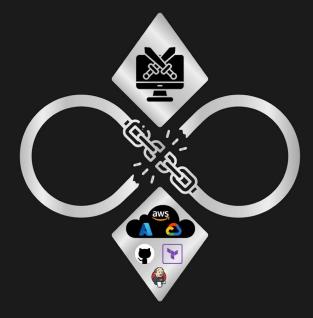




DO-RTA LAB ARCHITECTURE







CERTIFIED DEVOPS RED TEAM ANALYST [DO-RTA] GIVEAWAY



Thank You

For Professional Red Team / Blue Team / Purple Team / Cloud Cyber Range labs / Trainings please contact

support@cyberwarfare.live

To know more about our offerings, please visit: https://cyberwarfare.live