



สถาบันไอทีจีเนียส
IT GENIUS INSTITUTE

COURSE ID
OTC-12

COURSE OUTLINE · รายละเอียดหลักสูตรอบรม

CLOUD NATIVE / KUBERNETES

Kubernetes (K8s) Orchestration

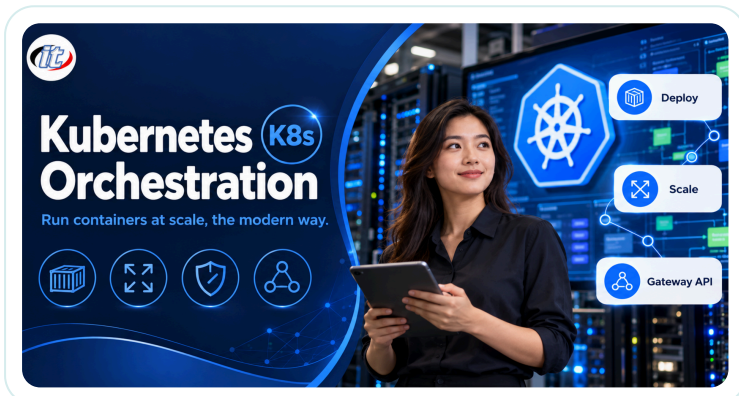
จัดการ Container บน Kubernetes 1.37 ด้วยแนวปฏิบัติปัจจุบัน

● 18 ชั่วโมง

● 3 วัน

● พื้นฐาน Linux และ Docker เบื้องต้น

● ระดับ Intermediate



ระยะเวลา: 3 วัน

18 ชั่วโมง (3 วัน วันละ 6 ชม.) · 09:00-16:00 น.



รูปแบบ: Classroom / Hybrid

Onsite หรือ Online ผ่าน Microsoft Teams · Public / In-house



รายละเอียดหลักสูตร

www.itgenius.co.th

Kubernetes เป็นแพลตฟอร์มมาตรฐานสำหรับรันแอปพลิเคชันแบบ Container ทั้งบน Cloud และในศูนย์ข้อมูลขององค์กร ช่วยให้ระบบดูแลตัวเองเมื่อมีปัญหา ขยายตามปริมาณงาน และอัปเดตเวอร์ชันได้โดยไม่ต้องหยุดบริการ แต่ตัวอย่างจำนวนมากที่พบทั่วไปยังใช้ Ingress NGINX ซึ่งโครงการหยุดดูแลไปแล้ว หรือยังเขียน YAML แบบเก่าที่ไม่ตรงกับแนวปฏิบัติปัจจุบัน

หลักสูตรนี้พาผู้เรียนใช้ Kubernetes 1.37 ตั้งแต่เตรียมเครื่องมือบน Mac และ Windows ทบทวนการสร้าง Container ที่พร้อมขึ้นคลัสเตอร์และออกแบบแอปตามหลัก Twelve-Factor App จากนั้นเจาะสถาปัตยกรรมภายใน ฝึกจัดการ Pod, Deployment, ConfigMap, Secret, Probe และ Autoscaling กำหนดสิทธิ์ด้วย RBAC เปิดบริการด้วย Service และ Gateway API จัดการข้อมูลถาวรด้วย Persistent Volume และ StatefulSet แพ็กแอปด้วย Helm 4 แล้วปิดท้ายด้วยการ Deploy ระบบ Microservices ครบวงจร (อบรม 3 วัน วันละ 6 ชั่วโมง รวม 18 ชั่วโมง ระดับ Intermediate)

เครื่องมือ

K8s Kubernetes 1.37

Dk Docker

Hm Helm 4

GW Gateway API

12F Twelve-Factor

ราคาค่าอบรม · PUBLIC / IN-HOUSE

9,500 บาท / ท่าน

ราคายังไม่รวมภาษีมูลค่าเพิ่ม (VAT 7%) · รวมเอกสารประกอบการอบรม Lab Guide ไฟล์ YAML และ Helm Chart ครบทุก Lab และ Certificate · รับจัด In-house ขอใบเสนอราคาได้

สมัครผ่านเว็บไซต์ ลด 10%



เพิ่มเพื่อน
LINE

☎ 02-570-8449

☎ 088-807-9770

💬 @itgenius

🌐 www.itgenius.co.th



01 วัตถุประสงค์ Objectives

- ติดตั้งและเตรียมสภาพแวดล้อม Kubernetes สำหรับฝึบบน Mac และ Windows ได้
- ออกแบบแอปพลิเคชันตามหลัก Twelve-Factor App ให้เหมาะกับระบบ Cloud Native
- อธิบายสถาปัตยกรรมภายในของ Kubernetes และการทำงานของแต่ละองค์ประกอบได้
- จัดการ Pod, Deployment, Rolling Update และ Rollback ด้วย kubectl ได้คล่อง
- แยกค่าตั้งด้วย ConfigMap และ Secret พร้อมกำหนด Probe และ Autoscaling
- เปิดบริการด้วย Service และ Gateway API แทน Ingress แบบเดิม
- เลือกใช้แนวทาง Stateless หรือ Stateful พร้อม Persistent Volume และ StatefulSet ได้ถูกต้อง
- แพ็กและติดตั้งแอปพลิเคชันด้วย Helm 4 ได้

02 กลุ่มเป้าหมาย Target Audience

- Software Developer และ Web Developer ที่ต้องนำแอปขึ้น Kubernetes
- DevOps Engineer และ System Administrator ที่ดูแลระบบ Container
- IT Infrastructure Engineer ที่กำลังวางแผนแพลตฟอร์ม Cloud Native ในองค์กร
- Solution Architect ที่ต้องออกแบบระบบบน Kubernetes
- ผู้ที่เคยเรียน Kubernetes รุ่นเก่าและต้องการอัปเดตเป็นแนวปฏิบัติปัจจุบัน

03 จุดเด่นของหลักสูตร Highlights

- อัปเดตเป็น **Kubernetes 1.37** แนวปฏิบัติปัจจุบันปี 2026
- **Gateway API แทน Ingress** พร้อมแผนย้ายจาก Ingress NGINX
- **ออกแบบแบบ Cloud Native** Twelve-Factor App และสถาปัตยกรรมภายใน
- **ระบบที่ทนและขยายได้** Probe, Autoscaling และ RBAC
- **ลงมือจริงทุกวัน** Lab คสมและ Workshop Microservices

04 ความรู้พื้นฐาน Prerequisites

- ใช้งาน Linux Command Line ได้ในระดับพื้นฐาน
- เคยใช้งาน Docker มาบ้าง เช่น สร้าง Image และรัน Container (มีบทวนให้ในหลักสูตร)
- เข้าใจพื้นฐาน Web Application และระบบ Network เช่น IP, Port และ DNS
- โน้ตบุ๊ก Mac หรือ Windows ที่มี RAM อย่างน้อย 16 GB และติดตั้ง Docker Desktop ได้ พร้อมสิทธิ์ติดตั้งโปรแกรม

05 สิ่งที่ต้องเตรียมและเครื่องมือ Setup & Tools

รายการ	รายละเอียด
อุปกรณ์	Notebook macOS หรือ Windows (WSL 2) RAM 16 GB ขึ้นไป เชื่อมต่ออินเทอร์เน็ต
คลัสเตอร์สำหรับฝึก	Docker Desktop พร้อม kind หรือ minikube สำหรับรัน Kubernetes บนเครื่อง
เครื่องมือหลัก	kubectl, Helm 4, k9s และ VS Code พร้อม Kubernetes Extension
Networking และ Storage	Gateway API Controller แบบโอเพนซอร์ส, cert-manager และ Local Path Provisioner
Monitoring	metrics-server สำหรับ Autoscaling และ kubectl top

หมายเหตุ: หลักสูตรนี้ปรับปรุงใหม่ปี 2026 จากฉบับเดิมที่สอนด้วย Ingress และ YAML แบบเดิม เนื้อหาเปลี่ยนเป็น Kubernetes 1.37 เพิ่ม Gateway API ซึ่งโครงการ Kubernetes แนะนำแทน Ingress ที่หยุดพัฒนาแล้ว พร้อมแผนย้ายออกจาก Ingress NGINX ที่หยุดดูแลตั้งแต่มีนาคม 2026 เพิ่ม ConfigMap, Secret, Probe, Autoscaling, In-place Pod Resize, RBAC, StatefulSet และ Helm 4 เครื่องมือทั้งหมดเป็นโอเพนซอร์สหรือใช้ฟรี ฝึกบนคลัสเตอร์ในเครื่องของผู้เรียน หากต้องการทดลองบน Managed Kubernetes ของ Cloud ใช้บัญชีของผู้เรียนเองซึ่งอาจมีค่าใช้จ่าย

06 ภาพรวมการเรียนรู้ 3 วัน Big Picture

วัน	หัวข้อหลัก
Day 1 Section 1-5	เตรียมเครื่องมือบน Mac และ Windows, บทวน Container ที่พร้อมขึ้น Kubernetes, หลัก Twelve-Factor App และสถาปัตยกรรมภายในของ Kubernetes 1.37
Day 2 Section 6-10	Pod, Deployment, Rolling Update และ Rollback, ConfigMap กับ Secret, Probe และ Autoscaling, Namespace, RBAC และ Resource Quota
Day 3 Section 11-15	Service, Gateway API แทน Ingress, Persistent Volume และ StatefulSet, แพ็กแอปด้วย Helm 4 แล้วปิดท้ายด้วย Workshop Deploy ระบบ Microservices



DAY
1

เตรียมพร้อมและแนวคิด Cloud Native

เนื้อหาการอบรม · Section 1-5

รวม
6 ชั่วโมง

1 Lab: เตรียมเครื่องมือบน Mac และ Windows

- ติดตั้ง Docker Desktop บน macOS และบน Windows ผ่าน WSL 2
- ติดตั้ง kubectl, kind หรือ minikube และ k9s
- สร้างคลัสเตอร์แบบหลาย Node บนเครื่องด้วย kind
- ตั้งค่า kubeconfig, Context และ Namespace เริ่มต้น
- Lab: สร้างคลัสเตอร์แรกและตรวจสอบสถานะ Node

2 นกวน Container ที่พร้อมขึ้น Kubernetes

- คำสั่ง Docker ที่ใช้บ่อยและวงจรชีวิตของ Container
- เขียน Dockerfile แบบ Multi-stage และรันด้วยผู้ใช้ที่ไม่ใช่ root
- ตั้งชื่อ Tag ของ Image ให้ติดตามเวอร์ชันได้ และเลือกฟังก์ latest
- Push Image ขึ้น Container Registry และโหลดเข้าคลัสเตอร์ kind
- Lab: สร้าง Image ของ Web API ตัวอย่าง

3 Kubernetes Overview

- Kubernetes คืออะไร แก้ปัญหาอะไรที่ Docker อย่างเดียวทำไม่ได้
- จุดเด่น: Self-healing, Auto-scaling, Rolling Update และ Bin Packing
- รอบการออกกลุ่มของ Kubernetes และการเลือกกลุ่มที่ยังได้รับการดูแล
- Managed Kubernetes บน Cloud เทียบกับคลัสเตอร์ที่ติดตั้งเอง
- ภาพรวม Ecosystem ของ CNCF ที่ใช้คู่กับ Kubernetes

4 The Twelve-Factor App

- Codebase, Dependencies และ Config แยกจากโค้ด
- Backing Services และ Build, Release, Run
- Processes แบบ Stateless, Port Binding และ Concurrency
- Disposability, Dev/Prod Parity, Logs และ Admin Processes
- Workshop: ตรวจสอบแอปตัวอย่างตาม 12 ปัจจัยและวางแผนปรับแก้

5 Kubernetes Architecture

- Control Plane: API Server, etcd, Scheduler และ Controller Manager
- Worker Node: kubelet, kube-proxy และ Container Runtime อย่าง containerd
- แนวคิด Desired State และ Reconciliation Loop
- บทบาทของ CNI, CSI และ CoreDNS ในคลัสเตอร์
- Lab: สำรวจองค์ประกอบของคลัสเตอร์ด้วย kubectl และ k9s



DAY
2

Workloads, Configuration และสิทธิ์

เนื้อหาการอบรม · Section 6-10

รวม
6 ชั่วโมง

6 Lab: Pod, Label และ kubectl

- เขียน Manifest แบบ Declarative และใช้ kubectl apply
- Label, Selector และ Annotation สำหรับจัดกลุ่ม Resource
- Pod หลาย Container, Init Container และ Sidecar Container
- ดู Log, exec เข้า Container และ Debug ด้วย kubectl debug
- Lab: รับ Pod ของแอปตัวอย่างและแก้ปัญหา Pod ที่ล้มเหลว

7 Lab: ReplicaSet และ Deployment

- ReplicaSet รักษาจำนวน Pod ตามที่กำหนดอย่างไร
- สร้าง Deployment และ Scale จำนวน Replica
- Rolling Update พร้อมกำหนด maxSurge และ maxUnavailable
- ดูประวัติและ Rollback กลับเวอร์ชันก่อนหน้า
- Lab: อัปเดตแอปโดยไม่หยุดบริการแล้วย้อนกลับเมื่อพบปัญหา

8 ConfigMap, Secret และ Resource

- แยกค่าตั้งด้วย ConfigMap ส่งเป็น Environment Variable หรือไฟล์
- เก็บข้อมูลลับด้วย Secret และข้อควรระวังเรื่องการเข้ารหัส
- กำหนด Requests และ Limits ของ CPU กับ Memory และระดับ QoS
- ปรับ CPU และ Memory ของ Pod ที่รันอยู่ด้วย In-place Pod Resize
- Lab: ย้ายค่าตั้งของแอปออกจาก Image ตามหลัก Twelve-Factor

9 Lab: Health Check และ Autoscaling

- Liveness, Readiness และ Startup Probe ต่างกันอย่างไร
- ติดตั้ง metrics-server และดูการใช้ทรัพยากรด้วย kubectl top
- Horizontal Pod Autoscaler ตาม CPU และ Memory
- PodDisruptionBudget เพื่อรักษาบริการระหว่างบำรุงรักษา Node
- Lab: ยิ่งโหลดทดสอบแล้วดู Pod ขยายและหดตามอัตโนมัติ

10 Namespace, RBAC และการแบ่งทีม

- แบ่งงานของแต่ละทีมหรือแต่ละ Environment ด้วย Namespace
- ServiceAccount, Role, ClusterRole และ RoleBinding
- จำกัดทรัพยากรด้วย ResourceQuota และ LimitRange
- Pod Security Admission และ securityContext ขั้นพื้นฐาน
- Lab: สร้าง Namespace ให้ทีม Dev พร้อมสิทธิ์และโควตา



DAY
3

Networking, Storage และ Workshop

เนื้อหาการอบรม · Section 11-15

รวม
6 ชั่วโมง

11 Lab: Services และ Service Discovery

- ClusterIP, NodePort และ LoadBalancer ใช้ในสถานการณ์ใด
- DNS ภายในคลัสเตอร์และการเรียกบริการข้าม Namespace
- EndpointSlice และการเลือก Pod ที่พร้อมรับ Traffic
- Headless Service สำหรับระบบที่ต้องรู้จัก Pod รายตัว
- Lab: เชื่อม Frontend, API และ Database ผ่าน Service

12 Lab: Gateway API และ Ingress

- สถานะของ Ingress ที่หยุดพัฒนา และการหยุดดูแล Ingress NGINX
- GatewayClass, Gateway และ HTTPRoute แบ่งหน้าที่ระหว่างกันอย่างไร
- Routing ตาม Host และ Path, แบ่ง Traffic ตามน้ำหนักสำหรับ Canary
- ทำ HTTPS ด้วย cert-manager บน Gateway
- Lab: ย้าย Ingress เดิมของแอปตัวอย่างมาเป็น Gateway API

13 Storage และ Stateful Workload

- Volume, PersistentVolume และ PersistentVolumeClaim
- StorageClass และ Dynamic Provisioning
- เมื่อไรควรเป็น Stateless และเมื่อไรต้องเป็น Stateful
- StatefulSet สำหรับฐานข้อมูลและ Cache พร้อมชื่อ Pod ที่คงที่
- Lab: รัน PostgreSQL ด้วย StatefulSet และทดสอบว่าข้อมูลไม่หาย

14 Lab: แพ็กแอปด้วย Helm 4

- Chart, Values และ Release ใน Helm
- สิ่งเปลี่ยนแปลงใน Helm 4 เช่น Server-side Apply เป็นค่าเริ่มต้น
- แยก Values ตาม Environment Dev, Staging และ Production
- Upgrade, Rollback และตรวจสอบสถานะ Release
- Lab: แปลง Manifest ของแอปตัวอย่างเป็น Helm Chart

15 Workshop: Deploy ระบบ Microservices

- ออกแบบระบบตัวอย่างที่มี Frontend, API หลายตัว และฐานข้อมูล
- Deploy ด้วย Helm พร้อม ConfigMap, Secret และ Probe
- เปิดบริการผ่าน Gateway API พร้อม HTTPS
- ทดสอบ Autoscaling, Rolling Update และการกู้คืนเมื่อ Pod ล้ม
- สรุป Checklist ก่อนขึ้น Production และแนวทางต่อยอดสู่ GitOps

07 เรียนจบแล้วได้อะไร Outcomes

- เตรียมคลัสเตอร์ได้เอง Mac, Windows และ kind
- เข้าใจสถาปัตยกรรม Control Plane และ Worker Node
- จัดการ Workload ได้ Deployment, Rollout และ Rollback
- ระบบทนและขยายได้ Probe, HPA และ RBAC
- เปิดบริการแบบใหม่ Service และ Gateway API
- ดูแลข้อมูลและแพ็กแอป StatefulSet และ Helm 4

08 สิ่งที่คุณเรียนจะได้รับ Deliverables

- เอกสารประกอบการอบรมฉบับสมบูรณ์ครบทั้ง 3 วัน และ Lab Guide ที่ละเอียด
- ไฟล์ Manifest YAML และ Helm Chart ของแอปตัวอย่างครบทุก Lab
- Checklist การย้ายจาก Ingress ไป Gateway API และ Checklist ก่อนขึ้น Production
- ประกาศนียบัตรจบหลักสูตร (Certificate of Completion) และช่องทางสอบถามวิทยากรหลังจบหลักสูตร

09 ข้อมูลผู้สอน Instructor

อ.สาธิต โคม ผู้สอนและผู้ปฏิบัติงานด้าน DevOps ที่ดูแลระบบ Container และ Kubernetes ในงานจริง สอนจากแอปตัวอย่างชุดเดียวที่ค่อย ๆ ย้ายขึ้นคลัสเตอร์ทีละขั้น อธิบายเหตุผลของแต่ละ Resource ให้เห็นว่าทำงานร่วมกันอย่างไร และให้ผู้เรียนลงมือแก้ปัญหาที่เจอด้วยตนเอง จนนำระบบขึ้น Kubernetes ได้อย่างมั่นใจ

ราคาค่าอบรม 9,500 บาท/ท่าน · สนใจ IN-HOUSE ขอใบเสนอราคา

9,500 บาท / ท่าน (ยังไม่รวม VAT 7%)

สมัครผ่านเว็บไซต์ ลด 10%

☎ 02-570-8449

📞 @itgenius

☎ 088-807-9770

✉ contact@itgenius.co.th



เพิ่มเพื่อน
LINE