DEVSECOPS WITH AWS

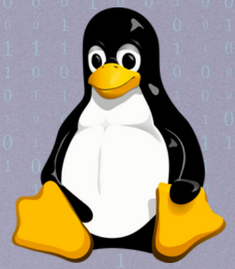
DEVSECOPS WITH AWS

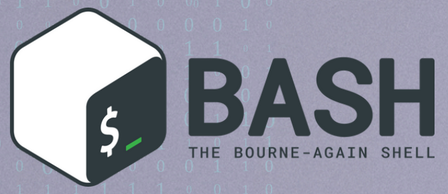
1. 100% Of The Training Is On AWS Infra
2. Training Starts From Basic To Advanced
3. Project Oriented Realtime Training From Day 1
4. Training on Micro-Services & Monolithic Project
5. Training on cut-edge Open Source Tech Stack
6. Life-time access to sessions
7. Daily Q&A + Interview Questions
8. Github Profile & 4 real-time CV's

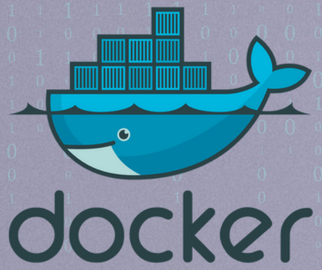
ABOUT TRAINER

1. An Experienced Professional with 12+ years of experience
2. Enterprise Experience & working as a Lead Architect
3. A professioal who is passionate about training & teaching
4. Trained more than 1500+ Professionals & Job-seekeers
5. Delivered more than 58 batches
6. 270+ 5 Star Reviews On UrbanPro & still counting

TOOLS COVERED IN THIS TRAINING

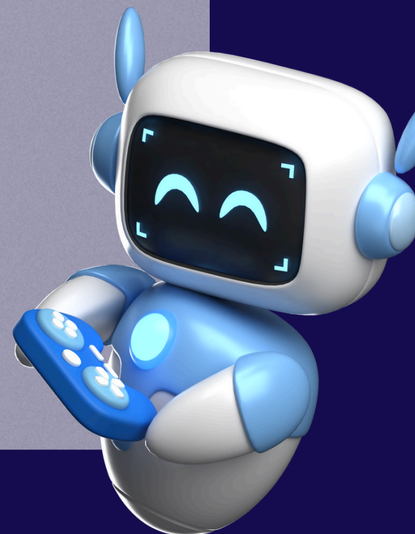






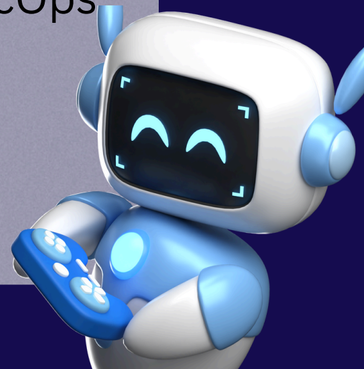
TOPICS COVERED IN aws

- What is AWS Cloud ?
- Why AWS Cloud ?
- Free Tier: Account Provisioning
- Cloud Business Models IaS vs PaS vs SaS
- Servers on AWS: EC2
- EC2 Instance Types: Spot vs On-Demand
- How to save 70% Billing on AWS?
- Regions vs Zones vs Wavelength Zones
- Virtual Private Cloud (VPC)
- Private & Bastion Network (VPN)
- Private, Public Subnets
- Authentication Modes on Cloud
- Key Less Authentication on AWS
- Load Balancers
- IAM Roles, Managed Policies, Trusted Policies
- OIDC Based Trust
- AWS Secret Manager
- S3 Buckets, KMS , Encyrption
- Document DB , RDS,
- ElasticCache, AmazonMQ
- EKS, Node Groups, Nodepools
- Plugin on EKS, Node Autoscaling
- Route 53, Regional HA, External DNS on EKS
- EC2 Instance Types: Spot vs On-Demand
- Elastic Container Registry
- Public & Private Registries
- Cloud Trail
- Amazon Certificate Manager
- Public & Private Hosted Zones
- AWS Opensearch
- Billing & Cost Management
- Auto Scaling Groups, Templates, Golden AMI’s.



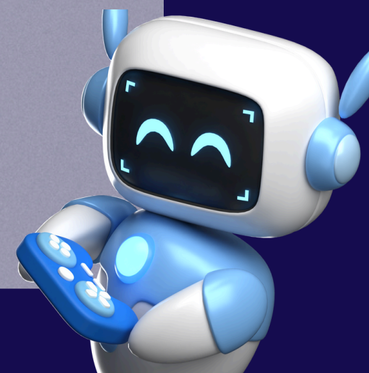
TOPICS COVERED IN

- What is Linux ?
- Why Linux ?
- Which Linux ? RedHat 9
- What is kernal & Opensearch ?
- Is Linux Free ?
- Can IT Survive without Linux ?
- Why Linux is Pre-requisite for Cloud & DevOps
- Which Linux do we learn ?
- Flavor vs Distro in linux?
- Is linux free ?
- How linux eco-system works ?
- Password -less vs AD based authentication
- File Management
- User Management
- Security Management
- Package Management
- RPM vs YUM vs DNF
- Configuring Private Repos
- Enabling Internet Connectivity to private machines.
- Process Management
- Security & Audit Management
- Log Management & Routing
- Auditd, Journald, Systemd
- File Management
- Day to day issues?
- How to make changes.
- Shortcuts & Day to day operations
- Options and commands needed for DevSecOps
- Storage Management
- File System Management



TOPICS COVERED IN BASH SCRIPTING

- Why Bash Scripting ?
- Bash vs Shell Scripting ?
- Problems that drove us to adopt bash scripting
- Why Bash is famous in Open Source ?
- Executing , Scheduling bash scripts
- SheBang Notation and Comments
- Pritning outputs and enabling color displays
- Declaring & Accessing variables
- Special Variables and harness the power of special variables
- Command & Arithmetic Substitute Variables
- Local Variables & Shell Variables
- Functions & Scope of variables in functions
- Functions in functions
- IF , IF Else, Else IF, Case Conditions
- For & While Loops
- String, Number & File Expressions
- Inputs , Redirects, SED, AWK, Streaming



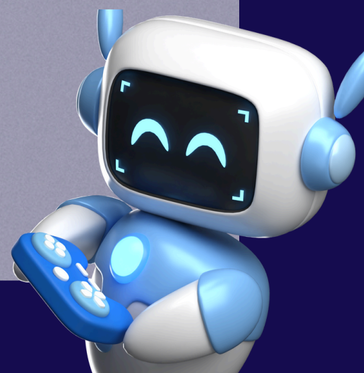
AUTOMATION USING BASH SCRIPTING

With Bash Learnings, we will automate 2 projects

- Monolithic Project
- Micro-services Projects

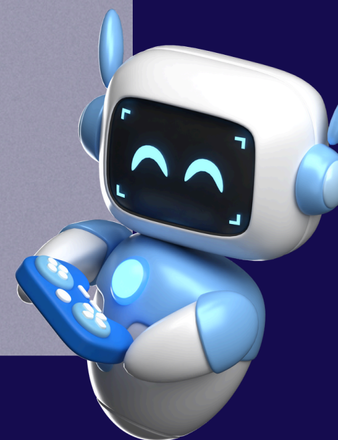
Standards

- DRY Based Approach.
- Extensive use of functions.
- 100% Automation with a single click on a button.
- Zero Keys based approach using IAM Roles.
- Versioning via GIT



TOPICS COVERED IN GIT

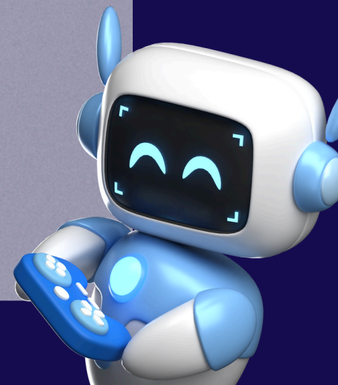
- Why GIT ?
- Why is GIT ?
- What is Version Control System ?
- GIT vs GitHub ?
- Internet without vs with GIT ?
- Can development work without GIT ?
- Repositories on GIT
- GIT basis operations
- GIT Bash Installation,
- GIT Commands,
- Local vs Remote Repositories
- Pull Requests vs Merge Request.
- Peer Reviews and approval process
- GIT Branching Strategy
- Importance Of GIT Commit Messages
- Cloning a repo, publishing commits to remote repo
- How can git can ignore the changes
- TAG’s & usage of Tags in the development
- git add fileName
- git commit -m “Msg”
- git push origin branchName
- git stash, git merge
- git checkout
- git clone, git pull, git push
- git fetch
- Cherry Picking Changes
- GIT Semantic Versioning
- Promotions Strategy using multiple branches
- TRUNK based development.
- App Release Strategy using GIT Repos
- Implementing GitOps
- Short Living vs Long Living Branches



TOPICS COVERED IN ANSIBLE

RED HAT® ANSIBLE® Automation

- Why is Ansible considered a top choice?
- What is Configuration Management, precisely?
- Is Ansible an open-source tool?
- Can you utilize Ansible at no cost?
- Challenges Associated with Shell Scripting
- How does Ansible compare to Chef and Puppet?
- Understanding Ansible's Push vs. Pull Mechanisms
- When should you opt for Push or Pull?
- Steps for Installing and Configuring Ansible
- Ansible Authentication Methods Explained
- Distinctions Between Collections and Modules in Ansible
- Best Practices for Ansible Inventory Management
- What exactly is a Playbook?
- An Introduction to Playbooks and Best Practices
- Managing Multi-Environment Inventories
- An Introduction to YAML Markup Language
- Fundamentals of YAML
- Variables at Play vs. Task Levels
- File vs. Shell Level Variables
- Variable Precedence
- Strategies for Keeping Playbooks DRY
- Roles and Hierarchy in Ansible
- Importing Roles from Tasks
- Introduction to Ansible Galaxy
- Utilizing Loops in Playbooks
- Implementing Conditions
- The Significance of Files, Templates, Meta, and Tasks Directories in Ansible Playbooks
- Debugging Techniques for Ansible Playbooks
- Encryption Methods in Ansible Playbooks
- Understanding Ansible Vault
- Decrypting Playbooks Using Ciphers
- Challenges Associated with Ansible Ad-hoc Commands



AUTOMATION USING ANSIBLE

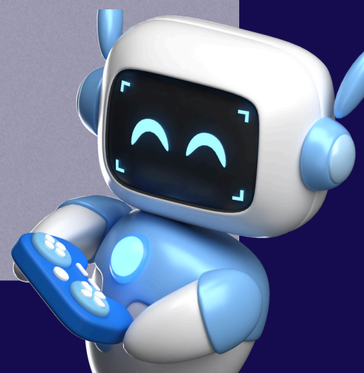
RED HAT® ANSIBLE® Automation

With Ansible Learnings, we will automate two projects:

- Monolithic Project
- Microservices Project

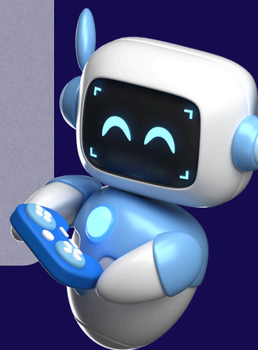
Standards include:

- A DRY (Don't Repeat Yourself) approach
- Extensive use of roles and task imports
- Complete automation with a single button click
- Ansible Vault for password encryption
- Vault integration with Ansible Playbooks
- Entire design centered around a multi-environment framework



TOPICS COVERED IN JENKINS & CI / CD

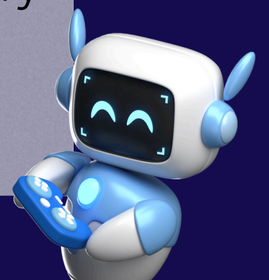
- Overview of CI/CD
- What is Jenkins?
- Reasons for Jenkins' Widespread Adoption
- Real-World Use Cases of Jenkins
- Installing and Configuring Jenkins
- Understanding Jenkins Architecture
- Implementing Master-Node Architecture
- Different Types of Jobs in Jenkins
- Comparing Free Style, Declarative, and Scripted Pipelines
- Job and Node Labeling
- Limitations of Free Style Jobs
- Declarative Pipeline: Agent, Stage, Steps
- Configuring Environment Variables & Parameters
- Job Triggers, Options, and Scheduling
- Creating Declarative Pipelines and Building Multiple Pipelines for CI/CD with Various Integrations.
- Creating CI/CD Pipelines
- Strategies for Keeping Pipelines DRY
- An Introduction to Jenkins Shared Library
- Getting Started with Groovy
- Developing Scripted Pipelines
- Advantages of Using Scripted Pipelines
- Incorporating Jenkins Shared Library into Pipelines
- Managing Jenkins Plugins
- Classic Jenkins UI vs. Blue Ocean UI
- Comparing Hudson and Jenkins
- Strategies for Ensuring Availability and Preventing Outages
- Post Section and Conditions in Pipelines
- Managing Secrets in Jenkins
- Integrating Jenkins with:
    - Sonarqube & SAST
    - Nexus
    - Vault



TOPICS COVERED IN TERRAFORM

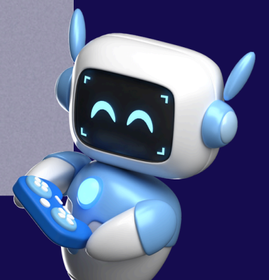
HashiCorp Terraform

- Introduction to Infrastructure as Code (IaC)
- Definition of IaC and Its Benefits
- Overview of Terraform
- Understanding Providers in Terraform: Why Are There Over 4,000?
- Introduction to HCL: Exploring Various Options in HCL v2
- Installing Terraform on RHEL 9
- Structure of Terraform: File Extensions and Standards
- Working with Variables, Inputs, Lists, and Maps
- Printing & Passing Outputs between modules.
- Backends in Terraform
- Differences Between Terraform .tfvars and .auto.tfvars, & TF Shell Variables
- Understanding Terraform Life Cycle Policies
- Executing Terraform Init & Terraform Plan
- Applying Changes with Terraform Apply
- Destroying Resources with Terraform Destroy
- Terraform State File Management
- TF Integration with DynamoDB
- Local State vs. Remote State
- Resource Blocks & Built-in Functions
- Conditions in Terraform
- Loops: for_each, count, count.index
- Utilizing Loops with count & for_each Arguments
- Inputs using Maps vs Lists
- Modules in Terraform
- Creating Modules to Maintain DRY Code
- Designing Terraform Infra Code for a Multi-Environment Approach
- Remote Modules vs. Local Modules
- Passing Outputs Between Modules
- Null, Remote, Local, and Destroy Provisioners
- Locals and Secret Management
- Terraform Best Practices Utilizing Real-Time Industry Use Cases
- Integrating Terraform with HashiCorp Vault



TOPICS COVERED IN DOCKER

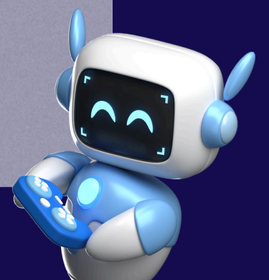
- Evolution of Computing Over the Last 30 Years
- Comparison of Physical and Virtual Machines .
- What Exactly is a Container?
- Is a Full OS Necessary to Run an Application?
- Can We Run Two Applications with Different Versions?
- An Introduction to Containers
- What is Docker and Why is it Popular?
- The Drawbacks of Docker: Is It Worth Learning?
- How Docker Simplified the Container Ecosystem
- Competitors of Docker.
- Understanding Containerd in Container Runtime
- Is There a Cost Associated with Using Docker?
- What is Container Runtime?
- Container Images: Public vs. Private
- Layers in Docker Imaging
- Learning the Basics of Docker
- Running Docker Containers from Public Repositories
- Volume Mounting and Port Forwarding
- Containerization
- Creating Our Own Docker Images.
- Multi-Stage Docker Imaging
- Best Practices for Docker Imaging.
- Developing Micro-Services Container Images
- Creating Container Images for Authentication with Vault.
- Various Docker Commands
- Utilizing Instructions: FROM, RUN, COPY, ADD, CMD, ENTRYPOINT, MAINTAINER
- Public Registries on Docker Hub and ECR
- Port Forwarding vs. Port Publishing
- Building Container Images with CI (Jenkins)
- Container Versioning
- Managing Container Vulnerabilities
- Enabling Profiles for Container Builds as Part of ECR Scans.
- Do we run containers in real-time ?



TOPICS COVERED IN KUBERNETES

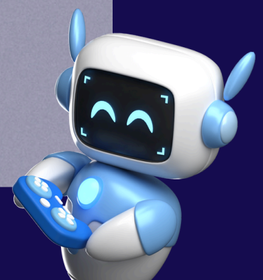
kubernetes

- What is Kubernetes or K8's ?
- What is Container Orchestration ?
- Introduction to kubernetes.
- Advantages of container orchestration using K8's.
- Set up sandbox K8 Cluster using Minikube
- Authentication to Kubernets Cluster, Kube Config.
- Kubernetes Architecture: Control Plane vs Data Plane.
- Kubernetes Manifest File: Intro to YAML
- Pod, Multi-Container Pod & Pod Life Cycle Management
- What is Pod ? Do we really run Pods ?
- Labels, Selectors & Annotations in Kubernetes
- Extract k8 resources using labels.
- Environment Variables, ConfigMaps, Empty Dir
- Secret Management using Secrets.
- Secret Injection To Pods using files.
- Init containers, Limits & Requests, Health Check Probes
- Sets in Kubernetes:
  - Deployment Set
  - Replica Set
  - Daemon Set
  - Statefule Set
- Pod priority and preemption.
- Automatic scaling of Pods utilizing HPA
- Taints, tolerations, and node affinity & anti-affinity.
- Services in Kubernetes: Cluster IP, Load Balancer, External Name, and Node Port.
- Benefits of Replica Set compared to Deployment Set.
- Configuring EKS on AWS with Terraform.
- EKS Node Groups and Node Auto Scaling.
- Namespaces and resource quotas.
- OIDC integration: EKS through IAM roles.
- AWS service authentication for Kubernetes workloads using IAM roles.
- Nginx Ingress Controller and Ingress resources.
- Installing Metrics Server for HPA and VPA.
- Service accounts, roles, RBAC, cluster roles, role binding, and access delegation.
- Monitoring EKS with Prometheus and Grafana.
- EKS log aggregation using the ELK stack.

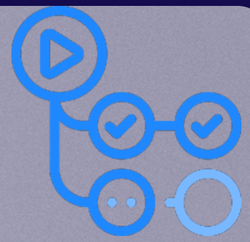


TOPICS COVERED IN SONARQUBE

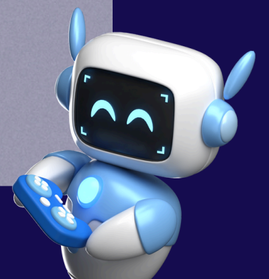
- What is Static Code Ananlysis
- Why SAST is needed ?
- SAST vs DAST
- What is Software Composition Analysis - SCA
- What is Sonarqube
- Continuous Monitoring using SonarQube
- SonarQube Integration To Pipeline
- Installing SonarQube on EC2 Instance
- install & configure sonarscanner
- enabling authentication to sonarqube from jenkins
- Submit Sonar-Scan to Jenkins using CLI
- Quality Qate vs Quality Profile
- Understanding Bugs, Vulnerabilities
- Understanding Code Smell, Code duplications.
- Design CI Pipeline to fail if the Quality Profile Conditions were not met.
- Sonarqube competitors & Sonarqube SaS Model



TOPICS COVERED IN GITHUB ACTIONS

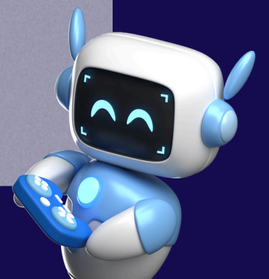


- Introduction to GitHub Actions
- What Exactly are GitHub Actions?
- Understanding Actions Architecture
- Are GitHub Actions Free?
- GitHub Actions vs. GitHub Workflows
- Exploring Events, Jobs, Steps, Runners, and Actions
- Setting Up GitHub Runners on EC2 Instances
- Crafting Our First Workflow with YAML
- Triggering Workflows via Push, Pull Requests, or Merges
- Environment Variables, Secrets, and Secret Management
- Managing Workflow Dependencies and Artifacts
- Utilizing Conditions, Loops, and Variables in Actions
- Integrating Workflows with EKS
- Developing CI Pipelines with GitHub Actions
- Best Practices for Using Actions in the Industry



TOPICS COVERED IN ELK STACK

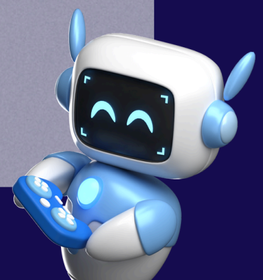
- What is Log Aggregation?
- An Introduction to Log Aggregation
- What is the ELK Stack?
- Installing Elastic Logstash, Kibana, and Automating the Installation
- How End Nodes Transfer Logs to ELK
- Structured vs. Unstructured Logs
- Transforming Unstructured Logs into Structured Logs Using GROK
- Creating GROK Expressions for Log Transformation
- Index Management and Log Labeling
- Aggregating Microservices Logs on ELK
- Designing Dashboards in Kibana
- Gaining Insights from Google’s Four Golden Signals
- Analyzing Project Error Rates, Traffic, Saturation, and Latency on Dashboards
- Cold Storage vs. Hot Storage
- Best Practices for Log Management



TOPICS COVERED IN VAULT

- What is Secret Management?
- Why is it crucial in DevSecOps?
- An Introduction to Hashicorp Vault
- Reasons Vault is the leading choice for secret management
- The seal-unseal procedure of Vault, including the unlock process using an OAUTH2 token
- How to install Vault on AWS EC2 infrastructure
- Configuring Vault and installing Vault CLI on your workstation
- Creating key-value based secrets
- Provisioning multiple secret engines based on environments and projects
- Data encryption and decryption processes
- Authentication and secret retrieval procedures
- Integrating Vault into CI pipelines and Jenkins
- Enabling secret management for Jenkins and Terraform through Vault
- Connecting EKS Cluster with Vault
- Establishing HTTPS connectivity for Vault using Certificates from AWS.
- Best practices for using Hashicorp Vault

HashiCorp
Vault



PROJECT ORIENTED REAL-TIME DEVSECOPS TRAINING

SUCCESS MANTRA

- DAY TO DAY PRACICE — 100% Important
- DAY TO DAY PARTICIPATION — 100% Important
- DOUBTS CLARIFICATION — 100% Important
- READY TO LEARN — 100% Important
- WILLINGNESS TO UNLEARN & LEARN — 100% Important