**Write Up Name :** Terraform

**Theory :**

M**odule No. 03**

1. **What is Terraform? Ans :**

* Terraform is our tool of choice to manage the entire lifecycle of infrastructure using [infrastructure as code.](https://www.fairwinds.com/blog/why-infrastructure-as-code-kubernetes) That means declaring infrastructure components in configuration files that are then used by Terraform to provision, adjust and tear down infrastructure in various cloud providers.
* Terraform is HashiCorp’s infrastructure as code tool. It lets you define resources and infrastructure in human-readable, declarative configuration files, and manages your infrastructure’s lifecycle. Using Terraform has several advantages over manually managing your infrastructure:
  + Terraform can manage infrastructure on multiple cloud platforms.
  + The human-readable configuration language helps you write infrastructure code quickly.
  + Terraform's state allows you to track resource changes throughout your deployments.
  + You can commit your configurations to version control to safely collaborate on infrastructure.

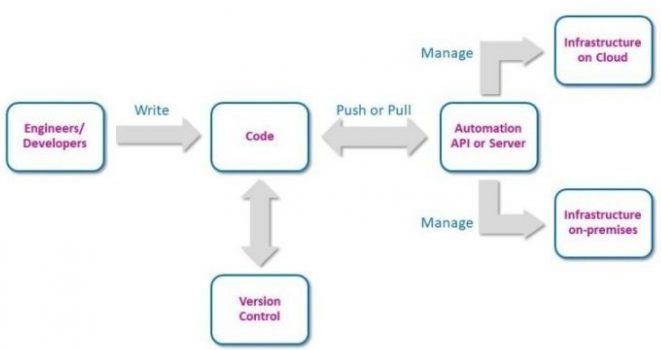
# What does infrastructure automation mean? Ans :

* Infrastructure automation is the process of reducing human interaction with IT systems by creating scripts or functions that are repeatable and can be used either by other software or oncommand.
* IT infrastructure automation tools can control IT elements including servers, storage, network elements, and operating systems (OS), with the goal of improving efficiency of IT operations and staff, with the goal of hands-off operation in most environments including both cloud and on-premises elements of hybrid IT environments.
* Cloud infrastructure automation tools greatly improve efficiency and speed of workload deployments and can help reduce costs by terminating unused virtual machines (VMs) and reclaim unneeded storage.

# What is the meaning of Infrastructure as code? Explain.

**Ans :**

* Infrastructure as code is a means by which engineers define the computer systems and managing the operations environment in the same way one does applications or other code for general release.
* With IaC, the operations infrastructure is managed by applying the same rules and strictures that govern code development instead of making manual configuration changes or using one-off scripts to make adjustments in the infrastructure.
* Most commonly, the engineers utilize a framework like Chef or Ansible, or Puppet to define their infrastructure.
* The type of tool is not important in this case as they all have similar facets, each allowing the engineer to define a computer system or network of computer systems necessary to run and support their code.
* Below is an illustration to depict IaC in action:



* Most often, IaC libraries are used in managing web applications, but the potential uses of IaC are endless. Modern applications are complicated and require web servers, load balancers, database servers, and queue processing systems, necessitating hours in provisioning, and managing these interlocking systems.
* This is a problem that IaC is designed to solve by enabling developers or administrators to create new environments to test code in minutes.
* This means improved and more frequent testing, and with frequent testing, the advantage is that the tiny tweaks that would emerge at a later stage are now identified. System administrators can document these tweaks and check them into source code.
* In the same way, configuration issues are easily identified when administrators need to change systems manually.

# Why do you think the software industry required something as a infrastructure automation?

**Ans:**

* Automation is a key component of IT orchestration, of driving IT efficiency, and of enabling an organization’s digital transformation. As IT organizations grow in complexity and size there are a limited number of resources to focus on infrastructure administration. Updates, OS patches, and the ordering, provisioning, configurating, and deploying of systems can present an unmanageable burden for IT were it not for a solid infrastructure automationframework.
* Every organization wants efficient, scalable, and reliable IT infrastructure. Automation tools increase agility, enhance employee productivity, and reduce security attack surfaces by eliminating manual errors though standardization of process and governance policies.
* Infrastructure automation mostly benefits those tasks that are self-contained, well documented, and tedious to perform manually, for example:

# Provisioning :

Infrastructure automation can reduce provisioning time for new VMs, storage, and networking from weeks or months to minutes or hours. This is especially valuable in today’s multi-cloud hybrid IT environments where automation and orchestration work hand in hand to ensure smooth deployment and operation regardless of a workload’s placement.

# Cost Reduction :

Server and VM sprawl can obfuscate the real costs of a given IT environment. Automation can highlight all cost components of virtual and physical IT infrastructure, enabling department chargebacks and pinpointing anomalies that could mean runaway or forgotten workloads. Without adequate cost containment, management and line of business users often view IT as a liability or cost center, rather than asset or profit center.

# Capacity Planning :

Both under-and over-provisioning impact performance organization-wide. Waste can occur due to a lack of standards in deployment or workloads. Infrastructure automation eliminates inconsistencies, thus reducing complexity and increasing standardization across the organization. This can help identify those areas where incorrect provisioning is impacting resource allocation and workload deployment, such as systems with more memory or CPU power than needed for a given workload or workloads.

# VM Sprawl :

The simplicity of spinning up new workloads or storage, whether in public or private clouds often leads to VM sprawl, the virtual cousin of server sprawl of years past. Cloud infrastructure automation tools help prevent VM sprawl by identifying workloads that are no longer used, automatically decommissioning those workloads and storage resources. This can help prevent public cloud sticker shock and improve on-premises IT utilization to save significant amounts of money in both capital and operating expenses.

# Configuration :

Your IT environment contains a variety of hardware and software. Consistently managing all of these by hand can lead to higher maintenance costs and inability to meet strict service-level agreements (SLAs). Automation gives you predictable and repeatable processes for managing configurations across operating systems to improve consistency, speed changes, and increase uptime.

# System maintenance :

IT teams do not usually grow in size at the same pace as the infrastructure they manage. Teams often struggle to maintain increasing responsibilities with their existing staffing levels. Automation helps teams manage large, complex IT infrastructures with their current staff. It can free your staff from tedious, time-consuming tasks and allow them to focus on more rewarding and strategic projects.