Experiment 11

Aim : To learn Software Configuration Management and provisioning using Puppet Blocks as a Case Study

Theory:

CASE STUDY –

What is Configuration Management?

Configuration management is the process of maintaining software and computer systems (for example servers, storage, networks) in a known, desired and consistent state. It also allows access to an accurate historical record of system state for project management and audit purposes.

System Administrators mostly perform repetitive tasks like installing servers, configuring those servers, etc. These professionals can automate this task, by writing scripts.

However, it is a difficult job when they are working on a massive infrastructure. The Configuration Management tool like a Puppet was introduced to resolve such issues.

About Puppet - *Puppet* is a system management tool for centralizing and automating the configuration management process. Puppet is also used as a software deployment tool. It is an open-source configuration management software widely used for server configuration, management, deployment, and orchestration of various applications and services across the whole infrastructure of an organization. It is written in Ruby and uses its unique **D**omain **S**pecific **L**anguage (DSL) to describe system configuration.

Features of Puppet Blocks :

- **Large installed base:** Puppet is used by more than 30,000 companies worldwide including Google, Red Hat, Siemens, etc. along with several universities like Stanford and Harvard law school. An average of 22 new organizations per day use Puppet for the first time.
- **Large developer base:** Puppet is so widely used that lots of people develop for it. Puppet has many contributors to its core source code.
- **Long commercial track record:** Puppet has been in commercial use since 2005, and has been continually refined and improved. It has been deployed in very large infrastructures (5,000+ machines) and the performance and scalability lessons learned from these projects have contributed in Puppet’s development.

- **Documentation:** Puppet has a large user-maintained wiki with hundreds of pages of documentation and comprehensive references for both the language and its resource types. In addition, it's actively discussed on several mailing lists and has a very popular IRC channel, so whatever your Puppet problem, it's easy to find the answer.
- **Platform support:** Puppet Server can run on any platform that supports ruby for ex: CentOS, Microsoft Windows Server, Oracle Enterprise Linux etc.

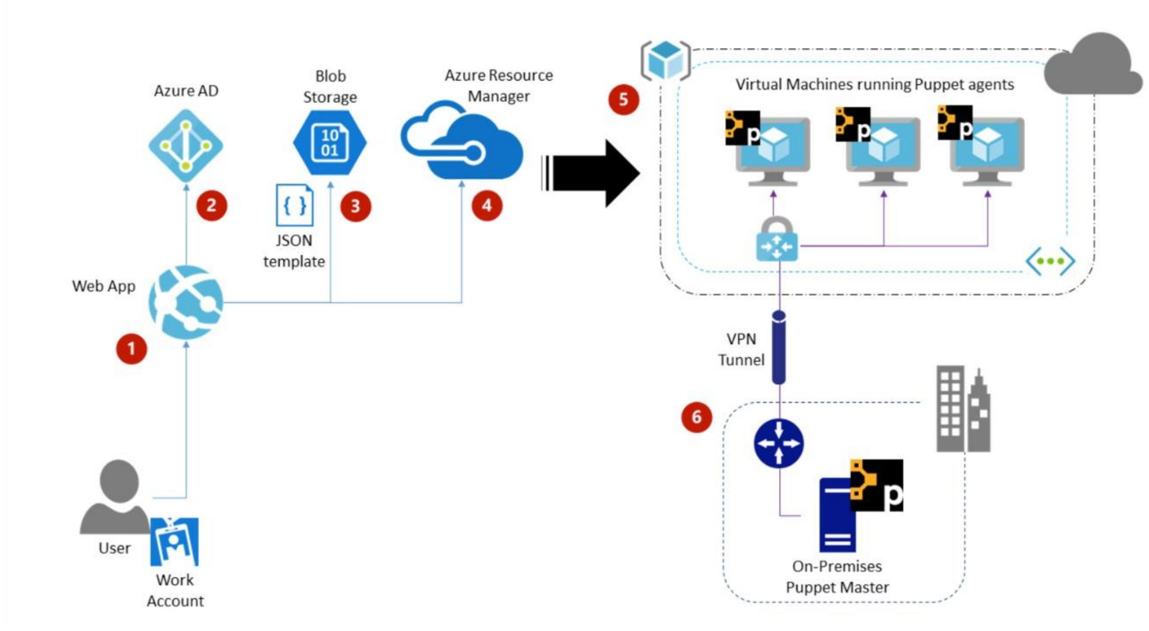
Real – Time Applications of Puppet Blocks as Software Configuring and Provisioning tool

1. **Zynga Game Developing Organization -** It is the world's largest social game developer. Zynga's infrastructure uses tens of thousands of servers in both public cloud and private data centers.

Here's how Puppet Blocks helped Zynga in its infrastructure repair :

- **Speed of Recovery** – The production operations team can rapidly deploy the right configuration to the right box. If a system gets inappropriately reconfigured Puppet will automatically revert it back to a last stable state, or provide the details necessary to manually remediate a system rapidly.
- **Speed of Deployment** – Puppet has provided significant time savings in the way the operations team delivers services for the gaming studios.
- **Consistency of Servers** – Puppet's model-driven framework ensures consistent deployments. According to *Mark Stockford, Vice President Production Operations*, Zynga *"It is evident that we have experienced time savings. The beauty of using Puppet is that it allows us to deliver consistent configurations across our servers in a short period every time."*
- **Collaboration** – Having a model-driven approach makes it easy to share configurations across the organization, enabling developers and operations teams to work together to ensure new service delivery is of extremely high quality. Over a dozen people from Zynga's team got trained in Puppet.

2. **Cloud applications with Azure -** Microsoft 's lead cloud services and web services provider . When members of the RISCO R&D team want a virtual cloud environment for development or testing, they request specific environments from IT, which must manually create them. These environments are multitiered and include various Azure services, so manual deployment includes creating these services, setting network configurations, and installing various modules on Windows and Linux virtual machines. This means that the time between an environment request to its deployment and availability to the requester can be as long as days and even a week, depending on the IT department's load. This has severe consequences on R&D processes.



Here’s how the team utilized Puppet blocks to automate cloud deployments in Azure and saved excess time .

- Self-service environment**.** The solution as a whole implements the “self-service environment” DevOps practice because it allows users to trigger the deployment of new cloud environments in a fully automated manner.
- Infrastructure as code. The use of Resource Manager templates allows RISCO to manage “infrastructure as code.” a DevOps practice that lets practitioners define and configure infrastructure (such as cloud resources) consistently while using software tools such as source control to version, store, and manage these configurations.

It was also important to the team to ensure that the deployment pipeline implemented by the self-service portal can be extended to serve a *continuous integration* and *continuous deployment* (CI/CD) process in the future. Automating the creation of new cloud environments is a key part of this process and extending the **self-service portal** to trigger such deployments via some build system (such as Jenkins) .

Conclusion : hence we studied the benefits of Puppet , real - time applications and we discussed in our case study where Puppet Blocks tremendously accelerated performance and boosted the infrastructure methodology of developers in **Zynga** and **Azure Microsoft**

references : 1 puppet.com
2. edeureka.com