DevOps Essentials DevOps Tools Notes

Introduction to DevOps Tools

The role of Tools in DevOps

* DevOps is NOT a set of tools
* But how can we achieve high speed of delivery while maintaining stability? Tools!
* The DevOps community has created a wide range of powerful tools
* Part of doing DevOps is identifying the tools you need and learning how to use them

The periodic table of DevOps Tools

* We can’t cover every tool in this course, so we will just briefly introduce you to some of the most popular ones.
* A way to explore some of the popular tools associated with DevOps: The Periodic table of DevOps Tools.
* <https://xebialabs.com/periodic-table-of-devops-tools/>

Tools for Build Automation and Continuous Integration

Build automation tools

* Build automation – Automated processing of code in preparation for deployment
* What tools you use for build automation usually depend on programming languages and frameworks

Examples

* Java – ant, maven, gradle
* Javascript – npm, grunt, gulp
* Make – widely used in Unix-based systems
* Packer – build machine images and containers

Continuous Integration Tools

* Continuous Integration – Continuously merging code into a single branch or mainline
* CI tools usually consist of a server that integrates with source control
* When source code is changed, the server responds by executing an automated build

CI tools

* Jenkins: Open source – fork of Hudson, Widely used, Java servlet-based
* TravisCI: Open source, build around GitHub integration, Executes builds in clean VMs
* Bamboo: Enterprise product by Atlassian, out-of-the-box integration with other Atlassian products like JIRA and Confluence.

Tools for Configuration Management

Configuration management tools

* Configuration Management – Managing and changing the state of pieces of infrastructure in a consistent and maintainable way
* Configuration management tools are a great way to implement IaC

Examples

* **Ansible**: Open Source
* Makes use of something called Declarative configuration. Declarative configuration is when you describe the state you want your configuration to be in and the tool determines what is necessary to get that environment into that state.
* YAML configuration files
* No control server needed.
* No agents needed installed, just uses python and ssh to connect to the host and make the changes it needs to make.
* **Puppet:** Declarative configuration, manage state through a UI.
* Custom modules use puppet domain specific language (DSL).
* Puppet pushes changes to clients using a control server and agents installed on clients.
* **Chef:** Procedural configuration- writing scripts to get the machine into the state you want.
* Uses an agent/server model.
* Uses a chef DSL.
* **Salt:** Declarative configuration
* Agent (minions) / server (master) – but can support agentless
* Uses YAML
* Support for event-driven automation

Tools for Virtualization and Containerization

Virtualization tools

* Virtualization – Managing resource by creating virtual rather than physical machines
* Hypervisor – Runs on bare metal and manages virtual machines
* Examples
  + VMWare ESX and ESXI
  + Microsoft Hyper-V
  + Citrix XenServer

Containerization

* Containers – Lightweight, isolated packages containing everything needed to run a piece of software
* Require fewer resources than VM’s – VMs contain an entire OS plus virtual versions of all the hardware
* Containers have the bare minimum needed to run the software
* **Docker –** docker is currently the leading container technology

Tools for monitoring

* Monitoring – Collecting and presenting data about the state and performance of applications
* There are different types of monitoring
  + Infrastructure monitoring – focuses on things related to infrastructure like CPU, or RAM
  + Application performance monitoring (APM) – focuses on performance and stability of individual parts of an application. Examples are like response times, and application logs

Infrastructure monitoring tools

* **SenSu:** Designed as a modern replacement for Nagios. Server/agent relationship. Agents push data to an AMQP broker.
* **NewRelic:** SAAS + agent, and has a wide variety of metrics and can also do APM monitoring.

Application performance monitoring tools

* **AppDynamics –** Collects data points about applications and presents it in a centralized dashboard.
* Code-Level diagnostics – able to identify performance issues at the code level. Has a Server/Agent relationship.

Aggregation and Analytics Tools

* Aggregation and Analytics are about collecting and monitoring data and doing something with it.
* Most monitoring tools come with some aggregation and analytics features
* **Elastic Stack-** pump data in and quickly create views to aggregate data and easily detect and diagnose problems.

Tools for Orchestration

* Orchestration – automation that supports processes and workflows, such as provisioning resources
* Lets you do things like:
  + Scale up and scale down applications on requests
  + Auto scale applications based on usage
  + Create self-healing systems by spinning down unhealthy nodes and replacing them with new ones

Orchestration tools

* **Docker Swarm:** Docker native orchestration tools. Orchestrates docker containers.
* **Kubernetes:** Open source and uses an orchestration server to control your resources. Manage containerized apps across multiple hosts.
* **Zookeeper:** Open source and founded by Apache. Can work alongside Kubernetes. Offers a centralized service registry that integrates with orchestration features.
* **Terraform:** Combines orchestration and IaC. Works well with other tools like Ansible. Works well with AWS and integrates with Kubernetes.