Automation and Orchestration

Automation

Automation is one of the reasons so many technical engineers are driven to a love of cloud technologies. In the case of AWS, there is a huge emphasis on API calls in order to configure the architecture. This permits automation of everything associated with AWS. Automation incorporates elements such as the following:

- Configuration templates
- Code deployment automation
- Self-healing infrastructures
- Reduction in the need for manual interventions
- Reduction in the potential for errors
- Lowered operating costs for Managed Service Providers (MSPs)

For many organizations relying on AWS nowadays approaching any challenge for their IT organization begins with the question, “How can we automate the solution?” Perhaps it is the corporate policy in your organization that you cannot use any of the default resources created for you in AWS. Sure, you could go into the Management Console and do lots of potentially error-prone mouse clicking, but things are so much easier (and more accurate) when you can automate such actions with a script.

Orchestration

A huge point of confusion for many engineers new to AWS and cloud technology is understanding the differences between cloud automation and cloud orchestration. One of the reasons why stems from the fact that the two terms are often used interchangeably, which is often incorrect. The differences between these concepts highlight a key challenge for teams looking to improve IT processes.

Let’s begin by reviewing automation. Automation describes a task or function accomplished without human intervention. So then what is orchestration? Orchestration describes the arranging and coordination of automated tasks, ultimately resulting in a consolidated process or workflow. Automation and orchestration go hand-in-hand, but note that they are technically different concepts.

With AWS, we like to create standard processes to spin up full environments to host new and exciting applications. We accomplish this by orchestrating many automated tasks. These might include the following:

- Automating new instances with Auto Scaling.
- Load balancing with automated ELB configurations.
- Deploying automation using a tool like CodeDeploy in AWS.
- Using Puppet scripts to automate the configuration of the OS.

While individually the tasks in the preceding list might be fairly simple to automate with the robust tools and capabilities of AWS, taken together these tasks can be very tricky to orchestrate. After all, these activities must occur in a particular order, under certain security

groups/tools, and be given roles and granted permissions. In other words, engineers must complete hundreds of manual tasks to deliver the new environment, even when the building blocks of that environment are automated. This is where orchestration is key.

Cloud orchestration tools, whether native to the IaaS (Infrastructure as a Service) platform or third-party software tools, enumerate the resources, instance types, IAM roles, and other resources required. Orchestration can also enumerate the configuration of these resources and the interconnections between them.

AWS engineers can use tools like CloudFormation to create declarative templates that orchestrate these processes into a single workflow so that the “new environment” workflow previously described becomes a single API call.

Well-orchestrated IT processes enable and empower continuous integration and continuous delivery, uniting teams in the creation of a set of templates that meet developer requirements. Such templates are in many ways living documents that embody the celebrated and popular DevOps philosophy.