Increasing Efficiency through Package Deployment Automation

A Technical Report submitted to the Department of Computer Science

Presented to the Faculty of the School of Engineering and Applied Science

University of Virginia • Charlottesville, Virginia

In Partial Fulfillment of the Requirements for the Degree

Bachelor of Science, School of Engineering

Carrick Hawes

Spring, 2020.

Technical Project Team Members

Carrick Hawes

On my honor as a University Student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments

Rosanne Vrugtman, Department of Computer Science

Increasing Efficiency through Package Deployment Automation

Carrick Hawes
Computer Science
The University of Virginia
School of Engineering and Applied Science
Charlottesville, Virginia USA
ch5vk@virginia.edu

ABSTRACT

SES Satellites, a software company that manages over 70 in-orbit satellites, decided to automate its inefficient, manual data package deployment process. I helped SES achieve automation by integrating various software into its deployment process. I stood up a Nexus repository manager to house packages after they are built by Bamboo. I leveraged an automation platform called Ansible to facilitate mass package distribution. I also incorporated a service called Docker to facilitate package deployment testing. SES's fully automated deployment system now allows packages to be built by Bamboo, automatically sent to Nexus, tested before deployment with Docker, and distributed by Ansible. SES employees can now write their own Ansible playbooks to automatically deploy packages to different groups of servers, test package deployments, and configure servers.

1. INTRODUCTION

Why spend manual effort on a task that can be automated? A study by Red Gate found that 74% of software companies surveyed in 2021 have adopted some automation practices (referred to in the article as DevOps) and this percentage has steadily increased from 47% in 2017 [1]. Moreover, in a 2017 McKinsey Global Institute report, it was found that automation can enable companies to improve performance by reducing errors and improving quality and speed [2]. SES is one of many tech companies recognizing the benefits of process automation. For my internship project, SES wanted to turn its package delivery process into a continuous pipeline by the automation tools Bamboo, Nexus, Ansible, and Docker.

Bamboo is an automation server used for continuous integration which allows developers to automatically build, document, integrate, and test source code in order to prepare an app for deployment [3]. It is not uncommon for software companies to use continuous integration and continuous deployment services like Bamboo. A recent report published by the Continuous Delivery Foundation (CDF) found that 44% of developers use either continuous integration or deployment tools (such as Bamboo), and this value increases to 60% for enterprise developers [4].

The next automation tool that SES wanted to incorporate into its package deployment process was Nexus. Nexus is a repository manager that organizes, stores and distributes artifacts needed for development. This is another relevant automation tool as Nexus is deployed at more than 100,000 organizations globally [5].

SES also wanted to use the automation software Ansible to facilitate package distribution and trigger testing scripts. Automation tools like Ansible are becoming increasingly popular among software companies. Over 16,349 companies started using Ansible in 2021 and many other companies use software similar to Ansible like Chef and Perforce [6].

The last automation software SES decided to use for my project is Docker. Docker is a service that offers container environments for testing (among other purposes). According to the DZone Container Trend Report, 2019 saw 20% of global organizations run container services like Docker and Kubernetes (a similar software) and 2023 expects this percentage to rise to 70% [7].

SES decided to automate its package deployment system with the software listed above. Recent years have seen an upward trend in popularity for each of these software as well as for automation as a whole. This adoption of automation practices indicates the growing importance and usefulness of automated processes when developing software.

2. RELATED WORK

Automation software like the ones I used in my internship are used to automate systems for many other companies, too. For example, HootSuite Media, Inc. uses Ansible for server deployment. Director of Technology at HootSuite, Beier Cai, said that before using Ansible, "[r]ebuilding a server relie[d] on limited documentation and mostly memory." After incorporating Ansible, server configurations could be stored in Ansible playbooks and then executed for easy server replication and deployment [8].

In addition, MetLife uses Docker to modernize its legacy applications. Wrapping legacy applications in containerized microservices allows for easier application adaptation and improvement. Using Docker has allowed MetLife to scale quickly when experiencing extra traffic and has increased MetLife's resource utilization, consolidating up to 70% of the company's VMs [9].

Furthermore, TD Bank uses Nexus to speed up the delivery process for its builds. TD Bank's adoption of Nexus was part of a company-wide transformation that moved the organization away from using the traditional Waterfall methodology and towards using Agile. Incorporating the Nexus repository manager has allowed TD Bank to "quickly and definitely trace, as well as account for, all of the components they were delivering to production" [10].

Bamboo is used by the Jet Propulsion Laboratory (JPL) (a subsidiary of NASA that tests new rocket technologies) in order to automate its Ensemble project. Ensemble is a platform for the development, integration, and deployment of mission operations software. When an issue arises, this software needs to be reconfigured or upgraded as quickly as possible since managing unmanned, in-orbit satellites is a high-stakes endeavor. Bamboo provides JPL with a "continuous deployment pipeline that delivers software updates to JPL's private cloud as fast as the Ensemble engineers can crank them out". Moreover, Bamboo is not the sole Atlassian product used by JPL; JPL also utilizes Atlassian's project management and documentation software, JIRA and Confluence [11]. This further aligns with the software from my internship, since JIRA and Confluence are also used by SES.

3. PROCESS DESIGN
3.1 Pre-existing System Challenges and Project Goals

In order to completely automate SES's package delivery process, the whole product life cycle needed to be considered. At SES, all products must go through the D2, T4, and S3 environments before reaching production. These environments stand for development, testing, and simulation, respectively. Figure 1 illustrates the product life cycle.
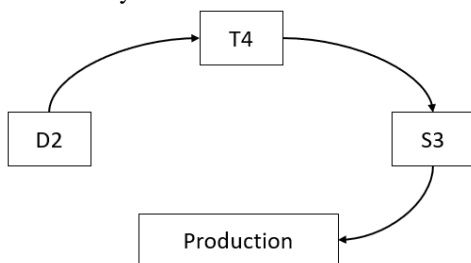

Figure 1: The product life cycle at SES.

Each of these environments contains several servers, all of which must be maintained through configuration and by installing and updating packages on them. In SES's pre-existing deployment process, packages were built by and stored in Bamboo, and Bamboo handled package distribution to the D2, T4, and S3 servers. However, Bamboo has limited deployment and testing capabilities. Consequently, SES employees had to manually trigger Bamboo package deployments to each server. Thus, this process did not facilitate easy, automatic server setup. In an ideal system, SES employees should be able to automatically install packages to different servers, packages would be stored in Nexus after being built, and there would be scripts available to automatically setup various servers.

3.2 Package Delivery Design Solution

The design solution for SES's pre-existing package deployment system consists of several software. Bamboo, Nexus and Ansible are leveraged to achieve process automation. Bamboo builds the packages and once built, those packages are sent to and stored in a Nexus repository. Simultaneously, Bamboo triggers an Ansible playbook that pulls those same packages from Nexus and distributes them to specific servers. Figure 2 demonstrates this process.
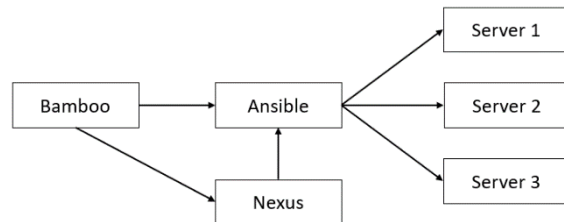

Figure 2: SES package delivery design solution.

Ansible is a better distribution tool than Bamboo because it has more deployment capabilities than Bamboo. However, Ansible cannot pull packages from Bamboo very easily, which is why a Nexus server needed to be stood up. Nexus provides Ansible with easier access to packages than Bamboo. I set up Nexus to act as an intermediary for Bamboo and Ansible, collecting builds from Bamboo and providing Ansible with a central location from which to pull and install packages on servers.

To create the Nexus server, I used a NGINX webserver as a reverse proxy to handle SSL certificate interactions. Setting up a reverse proxy was necessary for directing internet requests to the correct servers. Reverse proxy servers typically sit behind the firewall in a private network. Setting this up simplified SSL negotiations, since the reverse proxy server could host the SSL certificate and removed the need for encryption on the internal network. Figure 3 below illustrates the NGINX server's position in the network. After setting

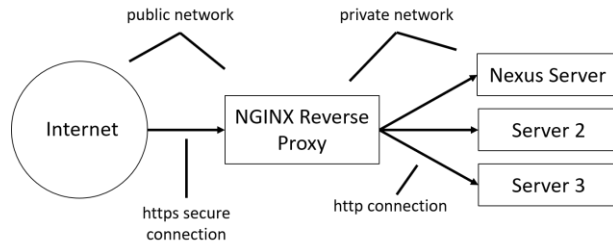up Nexus, I experimented with adding different types of repositories to it.

Figure 3: SES Nexus Network.

Once I learned how to operate Nexus, I then moved onto integrating Ansible with Bamboo. I downloaded an Ansible plugin for Bamboo which allowed Ansible playbooks to be triggered upon completing a build. I tested the functionality of this plugin by setting up Docker containers, writing Ansible playbooks to send packages to those containers, and triggering the playbooks I wrote with Bamboo. By this time, my internship was ending. Throughout my project, I documented my work and wrote instructions for how to use all the software I set up.

4. PROJECT OUTCOMES

My project resulted in the complete automation of SES's package delivery process. SES employees can now execute a product build in Bamboo that automatically builds, stores, tests and distributes the product to critical servers with full confidence that the package will deploy and function without error. Repositories can be added directly to Nexus or by building them in Bamboo and triggering their deployment to Nexus. 3 types of repositories can be added to Nexus: hosted, proxy, and group. Hosted repositories are hosted by Nexus and proxy repositories are remote repositories to which Nexus points. Group repositories are a group of repositories consisting of hosted and/or proxy repositories. While experimenting with Nexus, I was able to add repositories of each kind. For example, I added the CentOS-yum proxy repository which contains all the packages in the old repository (before Nexus) so that none of those packages are lost. I also added a hosted Docker registry. Docker registries contain pre-configured docker images. These images can be used to easily test Ansible playbooks. For instance, an Ansible playbook triggered by Bamboo after a build may contain commands to deploy to a docker image in the Nexus Docker registry. If this deployment is unsuccessful, commands can be added to the playbook so that an error message is returned and the deployment to critical servers is halted before any damage occurs. Figure 4 shows the various repositories I added to Nexus, including CentOS-yum and docker-hosted.
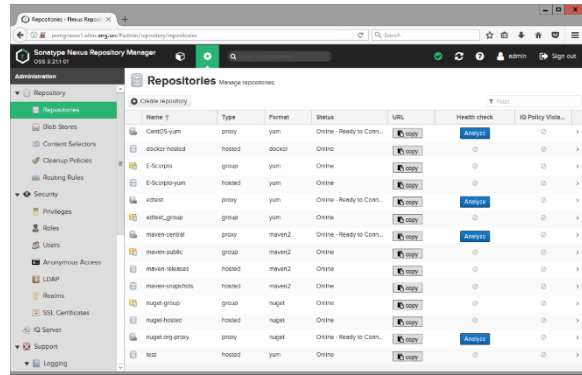

Figure 4: Nexus repository manager interface.

SES workers can also create their own Ansible playbooks to trigger package deployment to different groups of servers. If one wants to trigger the playbook they wrote after completing a Bamboo build, they will have to add the playbook to the list of tasks in the build. Figure 5 below shows an example of this.
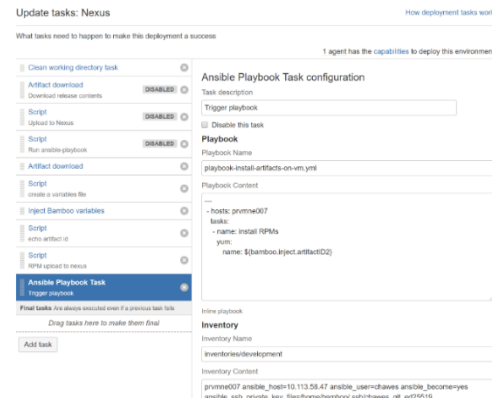

Figure 5: Bamboo interface.

In Figure 5, we see the Ansible Playbook task follows the Script task that sends the bamboo build (or artifact) to Nexus. Thus, a SES worker can simply write a playbook to deploy packages to certain servers then add a corresponding task to trigger the playbook's execution in Bamboo. Additionally, my project allows for the integration of company-wide changes that affect all servers since Ansible playbooks can be used to configure servers. For example, adding a non-root user account with an individual SSH key to all servers is possible using an Ansible playbook.

5. CONCLUSION

My project focused on getting products from the development stage of the product life cycle to the production stage. This included creating a pipeline that built products, stored them, tested them, and distributed them. In order to do this, I had to research the usability of different software, document my findings, and connect the various process steps to create a cohesive

deployment system. By building this pipeline, I was able to decrease the time and manual labor involved in SES's deployment process.

My project to automate SES's package deployment system was just one part of SES's plan to automate all of its systems. Even before my project, SES used several automation practices like Agile, Sprint meetings, and Kanban boards. SES additionally used various automation products offered by Atlassian, such as JIRA, Confluence, and Bamboo. SES is one of many tech companies that has embraced the automation movement. I mentioned earlier how companies like Hootsuite, MetLife, TD Bank, and the Jet Propulsion Laboratory are also transitioning towards automated practices. This shift is due to the increase in efficiency that automation creates. The automation of previously manual systems has been shown to reduce human errors (thereby improving product quality), boost production speed, and save companies money. Thus, projects such as the work I completed for SES end up being very valuable to companies.

6. FUTURE WORK

By the end of my internship, I completed the project I was working on at SES. A complete pipeline for the product life cycle had been created and now it just needs to be utilized. SES workers first need to familiarize themselves with the system I developed. Workers can watch the video presentation I created of my work that explains each step of the process. Next, SES employees should learn how to write their own Ansible playbooks and create some templates for common package distribution settings. Creating these templates minimizes the amount of editing workers need to do in order to distribute packages to their desired destinations. Similarly, playbook templates can be created for server configuration and testing environment settings, so that servers can easily be spun up when needed while requiring little-to-no configuration changes. I wrote a few simple playbooks that SES employees can view in order to get started with this process.

Moreover, SES workers should store additional packages in Nexus as desired. I added a few packages in Nexus to test that everything was working properly, but in order to use Ansible to its full capacity, all packages must be moved to this repository manager. Similarly, more Docker container environments should be added to the Docker registry in Nexus. This way, packages can first be distributed by Ansible to Docker containers and tested before they are sent to critical servers.

7. REFERENCES

[1] Pramod Sadalage, Kendra Little, Grant Fritchey, Steve Jones, Kathi Kellenberger, and Andrea Giardina. 2021. State of Database DevOps 2021. (2021). Retrieved February 6, 2022 from https://www.red-gate.com/solutions/database-devops/report-2021.

[2] James Manyika et al. 2021. Harnessing Automation for a future that works. (July 2021). Retrieved February 6, 2022 from https://www.mckinsey.com/featured-insights/digital-disruption/harnessing-automation-for-a-future-that-works

[3] Sahiti Kappagantula. 2019. Jenkins vs. Bamboo – Battle of the Best CI/CD Tools. (November 2019). Retrieved February 6, 2022 from https://dzone.com/articles/jenkins-vs-bamboo-battle-of-the-best-cicd-tools.

[4] SlashData. 2021. State of Continuous Delivery Report: 18 Months into the COVID-19 Pandemic. (December 2021). Retrieved February 6, 2022 from https://cd.foundation/cd-report-covid19/.

[5] Sonatype Inc. 2008. Nexus Repository: Software Component Management. (2008). Retrieved February 6, 2022 from https://www.sonatype.com/products/repository-pro.

[6] Slintel LLC. 2022. Ansible Configuration Management Tool: Top Customers and competitor details 2022. (2022). Retrieved February 7, 2022 from https://www.slintel.com/tech/configuration-management/ansible-market-share.

[7] DZone. 2021. DZone's 2021 Containers Trend Report. (2021). Retrieved February 7, 2022 from https://dzone.com/trendreports/containers-1.

[8] Dan London. 2014. Case study: Hootsuite. (October 2014). Retrieved February 13, 2022 from https://www.ansible.com/blog/ansible-app-deployment-hootsuite.

[9] Jenny Fong. 2017. Docker Enterprise Edition Lights a New Spark of Innovation within MetLife. (May 2017). Retrieved February 13, 2022 from https://www.docker.com/blog/docker-enterprise-edition-at-metlife/.

[10] Sonatype Inc. Customer Success Story - TD Bank. Retrieved February 13, 2022 from https://www.sonatype.com/customer-success/agile-transformation-at-td-bank.

[11] Atlassian. Atlassian visits Mars, Courtesy of NASA. Retrieved February 17, 2022 from https://www.atlassian.com/customers/nasa.