Project Title: Automate Operations using Ansible and RunDeck

Submitted by

Name: Tamara Foster
Title: Enterprise Systems Architecture Manager
Organization: Berkeley IT, Campus Applications and Data, Application Delivery, Application Security and Admin
Phone: (847) 212-5852
Email: tamarafoster@berkeley.edu

Project Team

Berkeley IT : Campus Applications and Data : Application Delivery : Application Security and Admin

Shiva Pachika – Project Lead and Systems Administrator - SIS
Yvonne Fang – Systems Administrator - SIS
Sri Chennamaneni – System Administrator - BFS
Kevin Feeley – Systems Administrator - BFS
Susana Parker – Systems Administrator – CalCentral

Berkeley IT : Campus IT Infrastructure : Data and Plaform Services : Operating Systems Support : Unix/Linux Operations and Services

Jeff Makaiwi – Unix Server Administrator

Project Summary

This project has greatly reduced the length of time needed for outages to apply maintenance to the Student Information System and the Berkeley Financial System, reduced the risk of errors inherent in doing steps manually, and provided easily understandable audit logs of the changes applied to each environment and server.

Project Timeline

Project Start: June 2020
Project Completion: April 2022

Project Information

The Berkeley Campus Solutions (BCS) and Berkeley Financial System (BFS) are Oracle PeopleSoft applications supported by the UC Berkeley IT Campus Applications and Data : Application Delivery : Application Security and Admin team. The CalCentral portal is a custom built portal that sits in front of BCS and is the entry point for all students, faculty and advisors at UC Berkeley for student and class information. The PeopleSoft Administrator team support yearly upgrades to the toolset as delivered by Oracle, quarterly security patches, and up to 6 application updates, legislative, and regulatory updates a year for each application.

From 2015 to 2021 the number of servers that need to be patched and updated grew from 40 to 150+. Each of these events require the production systems to have an outage, and until August of 2021 these updates were always done on the weekend because the outage window was significant. In addition to the production deployment, the team also have to apply these updates to all our development, test and quality assurance environments, and even with some scripting in place, the process was labor intensive and repetitive; there was a lot of manual configurations needed with each maintenance event, which is prone to error, and lastly, there were no audit logs to validate the work that had been done. These processes can be so labor intensive that they can easily consume the majority of an admin's time, leaving little room to make improvements or to adopt new features delivered by the tools, let alone to take on new responsibilities. This is in fact the state of many admin teams supporting these systems, and as the systems grow it becomes more and more challenging to find the capacity to introduce efficiencies and to innovate to make the processes faster.

The Solution

Over the past two years, the team has developed and implemented automated processes, using open-source software provisioning, configuration, deployment and scheduling tools named Ansible and RunDeck, for all of their operational processes. The development and implementation of automation resulted in a very significant reduction to our outage time during maintenance. In addition, we now have repeatable, reliable and verifiable processes that are re-used and that can easily be edited to adapt to changes in the environment. Some of the benefits achieved by this project are:

1. Significant reduction in outage window needed to apply maintenance to our systems.
2. SIS is now regularly patched on a weekday morning, and as of the last patch the system came back online in just over two hours and well before start of business.
3. BFS achieved reducing their outage windows from 3 days to 2 days for their most complex maintenance/release scenario. This result will support moving this effort away from a long weekend giving staff two long weekends back each year.
4. The admins are collaborating to architect their systems so that they can support each other.
5. The tools used provide readable audit logs that show the changes made to each system, as well as issues, in one central repository that can be accessed by all the admins.
6. The same code used for maintenance was leveraged during an emergency maintenance event in December allowing the team to patch all our supported servers including our development infrastructure in just over 8 hours resulting in a stronger security posture.

There are two major factors that allowed this team to successfully innovate to achieve this result in this short time:

1. Preparing the physical environment for automation: BCS and BFS teams both had a re-platforming events that required them to get new servers. They took advantage of setting up an architecture and structure to allow for automation.

2. Collaboration: While only one team member attended the conference where he learned about the tools, the team utilized working sessions on Zoom extensively and they planned and executed their plan incrementally, sharing learning and resources for learning as well as solutions.

The Story

In 2019 one of our admins attended the psadmin conference (psadmin.conf) and learned about how RunDeck (a runbook automation tool) and Ansible (an IT Automation tool), could be used to automate the maintenance of our growing portfolio of servers. Starting in June 2020, after re-platforming to RedHat Enterprise Linux 7, the SIS admins began to develop their first scripts for SIS. They had used the re-platforming project as an opportunity to re-architect the servers and directory structures to be consistent across the environments. This laid the foundation for automation and after a proof-of-concept period where they met in Zoom work sessions regularly, they were able to rapidly build out the automation of their patching. In late 2020 the BFS was re-platformed to RHEL7 and began the work to automate their patching. On July 28, 2021 we patched our Campus Solutions (BCS) production environment on a Wednesday morning for the first time. The confidence of the team to be able to bring the systems back online before Noon was a deciding factor in approving the first midweek outage, moving away from a Sunday patching event.

In April 2022, the team successfully used a the newly created dedicated Ansible Control VM to patch production BCS in just two hours from start to finish, finishing well before the start of business, and the BFS team completed their patching from start to finish in 8 hours, a full four hours sooner than expected. These numbers include the functional check out and pre and post processes which accounts for the variation in time between the systems. An example of the impact of this project is illustrated in the graph below which shows the time in minutes to apply the quarterly Critical Patch Updates (CPUs), which are security patches, from 2015 to 2021. From 2019 to 2020 we see a change from 320 minutes to 25-30 minutes.

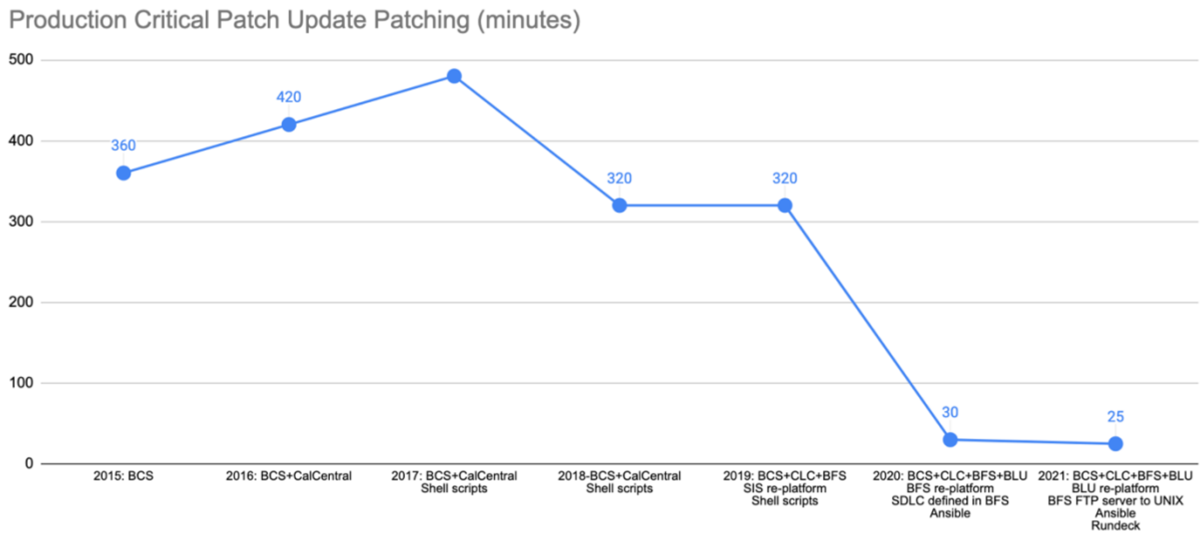


Chart: line chart showing the time in minutes required to apply a critical patch update from Oracle to the production environment for the student information system. In 2015 it took on average 360 minutes to apply a patch. As the system grew in size, the time increased to 480 minutes. After creating shell scripts and re-architecting to improve performance the time was reduced to 320 minutes in 2018 and 2019. After converting to Ansible and implementing RunDeck, the time to patch was reduced to 25-30 minutes in 2020 and 2021.

Because this project utilizes tools that are open source and well established, the results gained by using these tools at UC Berkeley can also be achieved by teams running systems like these at other UCs. The team has invested in learning and developing these tools and in aligning the systems so that the tools can be applied across our applications, and as a result we have a reliable, verifiable, and repeatable process that covers nearly all of our operational maintenance, including emergency maintenance patching. The team also continues to find other applications that can take advantage of automation, such as with CalCentral.

Lastly, the investment in implementing Ansible and RunDeck across our supported applications has resulted in the admins being able to schedule several tasks in advance and to cover for each other during vacations and other absences. This provides our team with more flexibility and coverage.

Alignment with FY22 Priorities

1. Operational Efficiency and Business Process Improvements
   - Application deployment and configuration during CPU Patch cycle - using Ansible, we were able to cut down from 3~4 hours to 20~30 mins for BCS

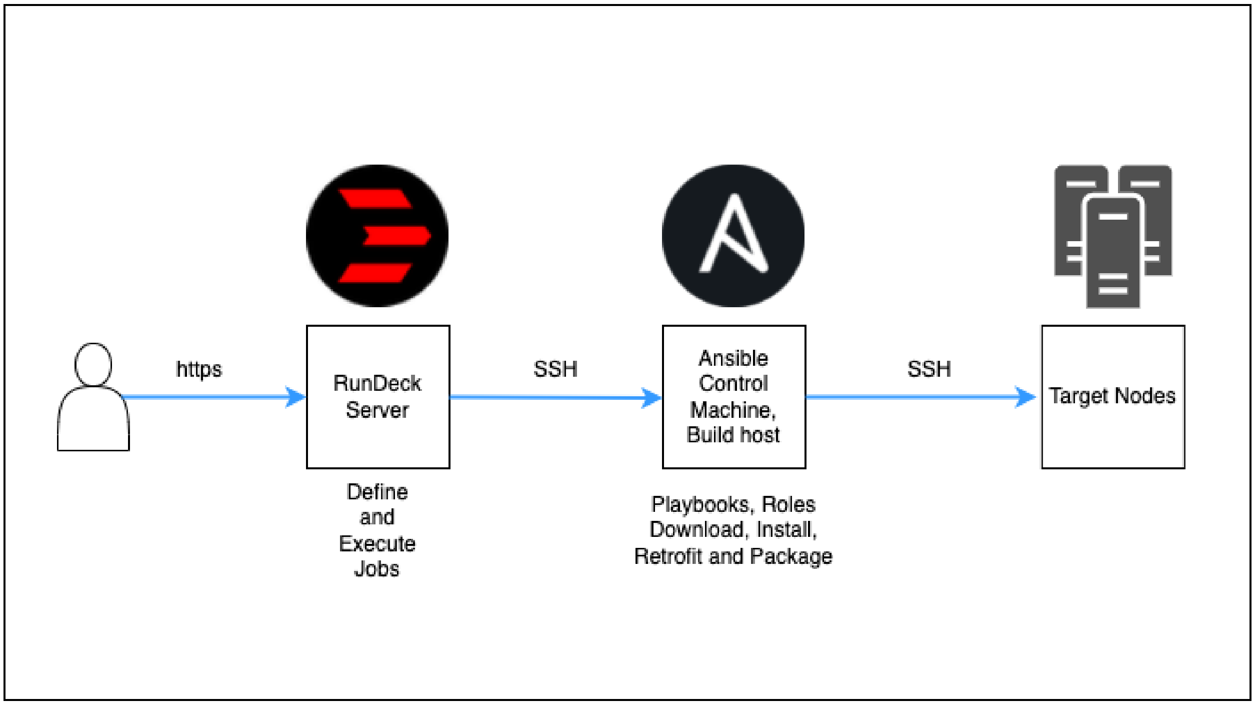
2. Process and Testing Automation
   - With RunDeck, we were able to setup Runbook automation, self-service operations, provide job scheduling visibility and activity logs
   - With Ansible, we were able to perform rolling updates and restart

3. Security Compliance
   - Using Ansible, we were able to meet the ISO timeline by patching within the 10-day time frame

4. Implement Service Management Principles
   - Cross-functional support, common tools, procedures and scripts provide standardization and streamline our processes across BCS, BFS, BLU and CalCentral systems

The Technical Architecture

RunDeck and Ansible are open source applications hosted on RedHat Enterprise Linux VM servers.


https
RunDeck Server
Define and Execute Jobs
SSH
Ansible Control Machine, Build host
Playbooks, Roles Download, Install, Retrofit and Package
SSH
Target Nodes

What’s next

The team continues to develop more automation and sharing their experiences across bIT and to the community. The team presented their work in our all staff meeting and at the Higher Education User Group (HEUG) Alliance 2022 conference in March 2022.