

# SE 492 – sdmay20-58

---

## K8s Checkmate

---

### Week 2 Report (1/27 – 2/13)

---

Client: Workiva

Faculty Advisor: Julie Rursch

Team Members:

- Daniel Brink - Delivery Manager
- Jacob Cram - Test Engineer
- Sean Sailer - Software Developer/Tester
- Alexander Stevenson - Project Manager/Team Lead
- John Young - Software Developer/Meeting Facilitator

### Past Week Accomplishments

---

#### Individual Contributions

Team Member Contribution Weekly

#### 27 Jan - 5 Feb

- Alex (27 Jan) [5 hours] - Was having an issue saving YAML files in a map so explored the idea of how to do it in Python. Made a new branch for that as it's only to see if that will be able to be used just in case it's needed. Currently it loads the yaml file, saves it to a map, then prints out the map.
- John (27 Jan) [4 hours] - Was looking into a solution that Jacob found that would parse helm charts with GO with having a generic type. I found that this solution would put the helm chart into a JSON file of strings. Worked with this route a little and realized it work, but forcing the code to work because of GO's limitations was causing some issues. Communicated with the team though and we decided that Python would be the better/easier option. -Jacob (27 Jan)[3 hours] - Did more research on the GO YAML parser, Found some issues but later decided with the team that Python would better satisfy our needs.
- Sean (30 Jan) [3 hours] - Researched the different python testing frameworks, and decided on which one best fits our needs and use cases
- Daniel (1 Feb) [2 hours] - Spent time setting up and re-familiarizing self with Python and PyTest.
- Daniel (2 Feb) [3 hours] - Set up a very basic logic checking function. As of writing, works with test keys and values, but does not work with checking images. Inital tests show correct behavior.
- Jacob (3 Feb) [4 hours] - Spent time attempting to setup Pyenv and virtualenv on my laptop because of our the switch to Python and ran into many issues. Once I figured this out, I now have a better understanding of how it works and did some personal testing with the utility.

#### 6 Feb - 13 Feb

- Sean (6 Feb) [3 hours] - Began setting up PyUnit test framework in our project allowing for automated testing of logic files. Additionally, began structuring tests for parsing data and storing it correctly
- Daniel (9 Feb) [3 hours] - Studying and comparing online examples of helm charts to better comprehend

what this project would be handling.

- John (12 Feb) [2 hours] - Looked at some examples of security checking policies online to get a better understanding on what a typical security protocol follows. Talked to Stevenson about whether or not to use online examples or security policies from Workiva's Helm charts.
- John (13 Feb) [5 hours] - Made some tests for checking the security policy based off the research I did.
- Jacob (13 Feb) [2 hours] - Worked on some adaptations to the first iteration of our helmparse that Alex originally created as a proof of concept.
- Alex Stevenson (11 Feb - 13 Feb) [7 hours] - Met with Eric, asked about charts to use to for testing and discussed checking image versions. Added policy parsing. Policy files are able to be parsed and stored in dictionaries. Worked on try to integrate the different parts. Added a requirements.txt.

Name	Bi-weekly Hours	Total Hours
Alex Stevenson	10	24
Daniel Brink	8	12
Jacob Cram	9	17
Sean Sailer	6	10
John Young	11	19

## Plans for Coming Week

---

- Figure out if the values must be saved to `values.yaml` or if they can be defined where they are used in the charts. If the latter is true, we will need to find a way to check all of the possible places they can be
- Find a way to parse the values from any sub-charts
- Figure out how templates play into this. Do we rerun the check on the templates? Do we have a list of allowed templates? How should go about dealing with them in a way that allows maximum flexibility to the end-user?
- Explore how we can utilize templates
- Parse a `policy.yaml` file and store the values
- Write methods to check the values from the policy against the chart
- Alert on any misconfigurations
- Figure out what to put in the policy files
- Figure out where users on the pods are defined

## Tasks

- Alex Stevenson is going to work on Phase 3, particularly the integration of the different parts. Going to try to have everything needed for the minimal viable product by the end of next week.
- John will work on implementing logic for checking and testing phase one and two
- Jacob will keep adapting the helmparse to better suit our needs and fit with Python best practice. Then work with the information that John and Daniel find about security issues and implement them into our system.
- Sean will finish setting up PyUnit and ensure all existing logic files have adequate test code coverage. Additionally, will help John on implementing logic for phase one and two.
- Daniel will work on value checking, testing and ensuring it functions properly with the types of data it needs to for phase three while keeping the team aware of commitments.