⎈ awesome-cka

A curated list for awesome resources needed to pass your [Certified Kubernetes Administrator (CKA)](#) exam.

📺 Background

The exam consists of the following, as of writing it is based on curriculum `1.14.1` ;

- 3 hours
- 24 questions
- Pass mark of 74%
- Remotely proctored
- Chrome browser plus an extension
- Government-issued non-expired ID
- Webcam and microphone
- Steady internet, preferably 5MB up/down.

The latest curriculum can always be found at https://github.com/cncf/curriculum.

During the exam you are allowed be allowed to open **one** tab apart from the browser based terminal. You can open *any* link under the [kubernetes.io](#) domain. Some links that will help during the exam are;

- https://kubernetes.io/docs/reference/kubectl/cheatsheet
- https://discuss.kubernetes.io
- https://kubernetes.io/docs/home/

If you are unsure of the spec or parameters of a yaml, always use `kubectl explain <resource>.<key>` .

🛠 Pre-requisites

Time is a big factor, you have **3 hours** to finish **24 questions**. That's **7.5 minutes per question** and they do get progressively harder, so no wasting time on the easier questions. If you're spending more than 7 mins on questions worth only one or two percent (each question does tell you the percentage it is worth), move on. You won't finish otherwise.

You need to be very comfortable with the following tools, so that you are not wasting time on non task orientated goals.

tmux

You only get one console, tmux and screen area allowed. This will allow you to split your single console into multiple windows. ie one master, two nodes. Try and use it in your daily workflow to get comfortable with the default settings.

- [ ] **Perform tmux deepdive** - https://thoughtbot.com/upcase/tmux - This course teaches you about tmux's pane and window management, session management, copy-paste, and more.

vi

- [ ] **Perform Onramp to Vim** - https://thoughtbot.com/upcase/onramp-to-vim - Get up and running with the world's best text editor. No Vim experience is required, but you'll be productive in no time (and blazing-fast, soon).

systemd

- [ ]

Read **Systemd Essentials: Working with Services, Units, and the Journal** and play - https://www.digitalocean.com/community/tutorials/systemd-essentials-working-with-services-units-and-the-journal - In this guide, we'll give you a quick run through of the most important commands you'll want to know for managing a systemd enabled server.

- ☐

  Read **How To Use Systemctl to Manage Systemd Services and Units** and play - https://www.digitalocean.com/community/tutorials/how-to-use-systemctl-to-manage-systemd-services-and-units - In this guide, we will be discussing the systemctl command, which is the central management tool for controlling the init system.

- ☐

  Read **How To Use Journalctl to View and Manipulate Systemd Logs** and play - https://www.digitalocean.com/community/tutorials/how-to-use-journalctl-to-view-and-manipulate-systemd-logs - In this guide, we will discuss how to use the journalctl utility, which can be used to access and manipulate the data held within the journal.

- ☐

  Read **Understanding Systemd Units and Unit Files** and play - https://www.digitalocean.com/community/tutorials/understanding-systemd-units-and-unit-files - In this guide, we will introduce you to the different units that systemd can handle..

kubectl

- ☐ Memorise **kubectl Cheat Sheet** and play - https://kubernetes.io/docs/reference/kubectl/cheatsheet/ - This page is an overview of the kubectl command. **NOTE:** This page can be referenced with your one other available tab.

These aliases will help save the precious time you have. Use these during your studies, so you are used to them on the day.

```
alias kc='kubectl'
alias kgp='kubectl get pods'
alias kgs='kubectl get svc'
alias kgc='kubectl get componentstatuses'
alias kctx='kubectl config current-context'
alias kcon='kubectl config use-context'
alias kgc='kubectl config get-context'
```

openssl/cfssl

- ☐ Memorise **OpenSSL command cheatsheet** and play - https://www.freecodecamp.org/news/openssl-command-cheatsheet-b441be1e8c4a/ - When it comes to security-related tasks, like generating keys, CSRs, certificates, calculating digests, debugging TLS connections and other tasks related to PKI and HTTPS, you'd most likely end up using the OpenSSL tool.

📚 Study

This is a list of the bare minimum resoures necessary to try and pass the exam.

Read

- ☐ Read **Kubernetes in Action** - https://www.manning.com/books/kubernetes-in-action - Kubernetes in Action is a comprehensive guide to effectively developing and running applications in a

Kubernetes environment. Before diving into Kubernetes, the book gives an overview of container technologies like Docker, including how to build containers, so that even readers who haven't used these technologies before can get up and running.

Additional Reading

> ***Kubernetes Up and Running*** *by Kelsey Hightower, Brendan Burns, Joe Beda -* [*https://www.amazon.com/Kubernetes-Running-Dive-Future-Infrastructure/dp/1491935677*](https://www.amazon.com/Kubernetes-Running-Dive-Future-Infrastructure/dp/1491935677)

> ***DevOps with Kubernetes*** *by Hideto Saito, Hui-Chuan Chloe Lee, Cheng-Yang Wu -* [*https://www.amazon.com/DevOps-Kubernetes-Accelerating-container-orchestrators-ebook/dp/B0711KDB8N/*](https://www.amazon.com/DevOps-Kubernetes-Accelerating-container-orchestrators-ebook/dp/B0711KDB8N/)

> ***The Kubernetes Book*** *by Nigel Poulton -* [*https://www.amazon.com/Kubernetes-Book-Version-January-2018-ebook/dp/B072TS9ZQZ*](https://www.amazon.com/Kubernetes-Book-Version-January-2018-ebook/dp/B072TS9ZQZ)

Do

- ☐

  Do **all** the tasks on https://kubernetes.io/docs/tasks/ – This section of the Kubernetes documentation contains pages that show how to do individual tasks. A task page shows how to do a single thing, typically by giving a short sequence of steps.

- ☐

  Do **kelseyhightower/kubernetes-the-hard-way** - https://github.com/kelseyhightower/kubernetes-the-hard-way - This tutorial walks you through setting up Kubernetes the hard way. This guide is not for people looking for a fully automated command to bring up a Kubernetes cluster.

Courses

If you prefer doing course instead of reading books, here are some courses for you to go through.

- ☐

  Do **Cloud Native Certified Kubernetes Administrator (CKA)** course - https://linuxacademy.com/linux/training/course/name/cloud-native-certified-kubernetes-administrator-cka - This course prepares you for the Certified Kubernetes Administrator (CKA) exam by the Cloud Native Computing Foundation. You will learn how all of the components of a Kuberenetes cluster work together, how to monitor all components of a cluster, and how to build your own Kubernetes cluster from scratch. We will also cover networking, deploying applications, scheduling pods, logging, and a whole lot of practice in the command line.

- ☐

  Do **Certified Kubernetes Administrator (CKA) with Practice Tests** course - https://www.udemy.com/certified-kubernetes-administrator-with-practice-tests/ - Prepare for the Certified Kubernetes Administrators Certification with live practice tests right in your browser.

- ☐

  Do **Kubernetes Deep Dive** course - https://acloud.guru/learn/kubernetes-deep-dive - Everything you need to know to start deploying and managing cloud-native applications on Kubernetes in the real world.

💼 The Coal Face

With the CKA being a practical exam, you will need significant experience working with clusters; setting up clusters, fixing clusters, and backing up clusters. Here are several options to get started and more advanced ones as well.

- ☐

  Use **Katacoda's Kubernetes Playground** - https://www.katacoda.com/courses/kubernetes/playground - This is a Kubernetes playground, a safe place designed for experimenting, exploring and learning Kubernetes. The playground has a pre-configured Kubernetes cluster with two nodes, one configured as the master node and a second worker node.

- ☐

  Use **minikube** - https://github.com/kubernetes/minikube/ - minikube implements a local Kubernetes cluster on macOS, Linux, and Windows.

- ☐

  Use **MicroK8s** - https://microk8s.io/ - MicroK8s is a CNCF certified upstream Kubernetes deployment that runs entirely in your workstation. Being a snap it runs all Kubernetes services natively (i.e. no virtual machines) while packing the entire set of libraries and binaries needed. Installation is limited by how fast you can download a couple of hundred megabytes and the removal of MicroK8s leaves nothing behind.

- ☐

  Use **Google Kubernetes Engine** - https://cloud.google.com/kubernetes-engine/docs/quickstart - Kubernetes Engine is a managed, production-ready environment for deploying containerized applications. It brings our latest innovations in developer productivity, resource efficiency, automated operations, and open source flexibility to accelerate your time to market.

🎯 Practice Test

This following here is great for a testing scenario. It gives you 24 questions to answer, a terminal, and a timer. You bring the cluster and kube config. It's highly suggest running through this to get comfortable with the test environment as this is the best replica for the test environment out there. It uses the same terminal emulator that the test uses, https://github.com/liftoff/GateOne.

- ☐

  Do **github.com/arush-sal/cka-practice-environment** - https://github.com/arush-sal/cka-practice-environment - A sample lab test environment to help in preparation of CKA certification.

- ☐

  Do **github.com/stretchcloud/cka-lab-practice** - https://github.com/stretchcloud/cka-lab-practice - A set of exercises that will help you to prepare for the Certified Kubernetes Administrator exam.

- ☐

  Read **github.com/dgkanatsios/CKAD-exercises** - https://github.com/dgkanatsios/CKAD-exercises - A set of exercises to prepare for Certified Kubernetes Application Developer exam by Cloud Native Computing Foundation, some overlap with the above but with more content.

🌏 Extra Reading

In case you yearn for more information about the exam and it's topics, here is another brilliant list of curated resoures that you will find handy.

- ☐ Read **CKA Resources** - https://gist.github.com/strongjz/4c9ad30a12ab715ae94cf72d0e7bbc30

A big thank you has to go to James Strong for his awesome curation of resources for the CKA.