**FINAL: Intel News Article #3**

**10/06/22**

**Words: 1,784**

**Tactical Edge Reference Architecture**

*Deploying at the Edge with Kubernetes*

*By: Andy Clemenko, Field Engineer, Rancher Government Solutions*

[andy.clemenko@rancherfederal.com](mailto:andy.clemenko@rancherfederal.com)

*Twitter:* [@clemenko](https://twitter.com/clemenko?ref_src=twsrc%255Egoogle%257Ctwcamp%255Eserp%257Ctwgr%255Eauthor)

Rancher Government Solutions (RGS) field engineers get asked all the time about how to deploy at the edge. Search the internet for "edge computing" and you’ll get more than *300 million* results that all say something different. This tells us two important things: 1) there is enormous interest in the topic; and 2) there is also enormous confusion about it.

This piece is designed to dispel that confusion and outline exactly how we at RGS deploy at the tactical edge leveraging Kubernetes.



First, let’s define the tactical edge.

## What is the Tactical Edge?

In 2012, the Software Engineering Institute and Carnegie Mellon University partnered on a piece on [Cloud Computing at the Tactical Edge](https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=28021).

From the abstract: “*Handheld mobile technology is reaching first responders, disaster-relief workers, and soldiers in the field to aid in various tasks, such as speech and image recognition, natural-language processing, decision making, and mission planning.”*

Fast-forward 10 years to today and it is amazing to see the amount of compute that can fit inside a coffee can. Case in point: ASROCK Industrial has a very powerful [4X4 BOX-5800U](https://www.asrockind.com/en-gb/4X4%20BOX-5800U) computer in a package that is 4 inches by 4 inches by 2 inches. Not only is it tiny, but its power consumption is low enough to run on batteries. Compute has gotten both powerful enough and small enough to run big applications from the palm of your hand.

At Rancher Government, we define the tactical edge as: A small case or kit that contains compute power and that is easily portable. The number and size of the nodes can vary. Being portable is the key characteristic.

Using this definition, we can start to think about real world applications of such a cluster, box, backpack or kit. The idea of "moving computation to where the data lives" can be applied. A lot of our work involves helping organizations process data in the field and then distill it into more meaningful data to be sent back upstream.

### Important Considerations for Computing at the Tactical Edge

For all its potential and enabling power, there are two important considerations for computing at the tactical edge. The first one involves power, space, and cooling. And it’s not surprising. These are challenges for even the biggest data centers in the world.

When looking for edge hardware, pay attention to the [Thermal Design Power](https://en.wikipedia.org/wiki/Thermal_design_power), or TDP for short. This will indicate how much power is required for that device. Remember that the power requirements will increase when the device is under a heavy load. Calculating the combined TDP for an edge kit is important to understanding how long a battery will last, or how much power supply is needed. For reference, the kit below has a combined TDP of roughly 185 watts. Computing at the edge requires sufficient power, and enough space to allow for cooling.

The second major consideration involves manageability, specifically how to manage the operating system and applications far away from "home.” This is where Kubernetes and GitOps can dramatically increase the velocity of deployment and manageability. Ideally, teams will want a lightweight Kubernetes distribution and management layer since not all Kubernetes distributions can operate in a smaller compute envelope. The Container Journal recently highlighted a lot of the advantages of leveraging Kubernetes at the Edge.

## Hardware

Once power, space, cooling, and manageability have been assessed, teams can start thinking about a tactical edge architecture.



Boy, do we have many choices of hardware these days! X86 or Arm? Intel or AMD? My favorite approach for picking hardware is to look at the applications that are needed. We call this approach "working backward.” Once the total amount of compute required is calculated, we can start looking at the amount of CPU that will be needed. Adding the application CPU and Memory together with the Kubernetes and management overhead we get the "compute envelope” – meaning, the total amount of CPU cores and memory. Of course, there are other components, like networking, that are worth looking at as well.

For this reference architecture, we have chosen three [ASROCK 4X4 BOX-5800U](https://www.asrockind.com/en-gb/4X4%20BOX-5800U) boards. There is a good balance of cores, memory, storage and TDP. Each board has eight cores and supports up to 64gb of ram. For storage, there is NVME (PCIe Gen3x4) support and a SATA3 port. As for power, we were able to use a single 150w power supply for all three boards. Each board has a TDP of 60Watts. Under heavy load there were no issues at all. The boards also have multiple NICs, including a 2.5gb one.

To maximize the portability, we added a [Gl.inet Beryl Travel Route](https://www.gl-inet.com/products/gl-mt1300/). The router is great for extending the connectivity to additional devices like a laptop. The router also has a "repeater" function for Wifi – meaning, all the nodes reach the internet for updating and initial loading of software. And of course, an internal DHCP server. As a side note, some of the Gl.iNet routes also have WireGuard (VPN) capabilities.

One last piece to this architecture is a [Netgear GS105](https://www.netgear.com/business/wired/switches/unmanaged/gs105/) five-port one-gigabit ethernet switch. The switch provides communication between the nodes. The switch could easily be upgraded to 2.5 gigabits if needed. However, if the case only had two nodes, the GL.iNet router would be able to handle all the internal traffic.

## Operating System

Similar to the array of choices there are in the hardware realm, there are a number of choices when it comes to operating systems. [Ubuntu](https://ubuntu.com/) is a great choice for many scenarios. It is built on Debian which has been around for decades. [Rocky Linux](https://rockylinux.org/) is another great choice. Rocky is the new Centos. Rocky is built from RHEL with all the enterprise security and stability built in. We have a few guides that talk about using Rocky as a secure foundation for RKE2. For this guide, Rocky for the win! And yes, please leave SElinux enforcing. Use installation method of your choice. If you have PXE infrastructure in place, use it. For cluster in the pictures, we used a usb-c thumb drive.

## Software

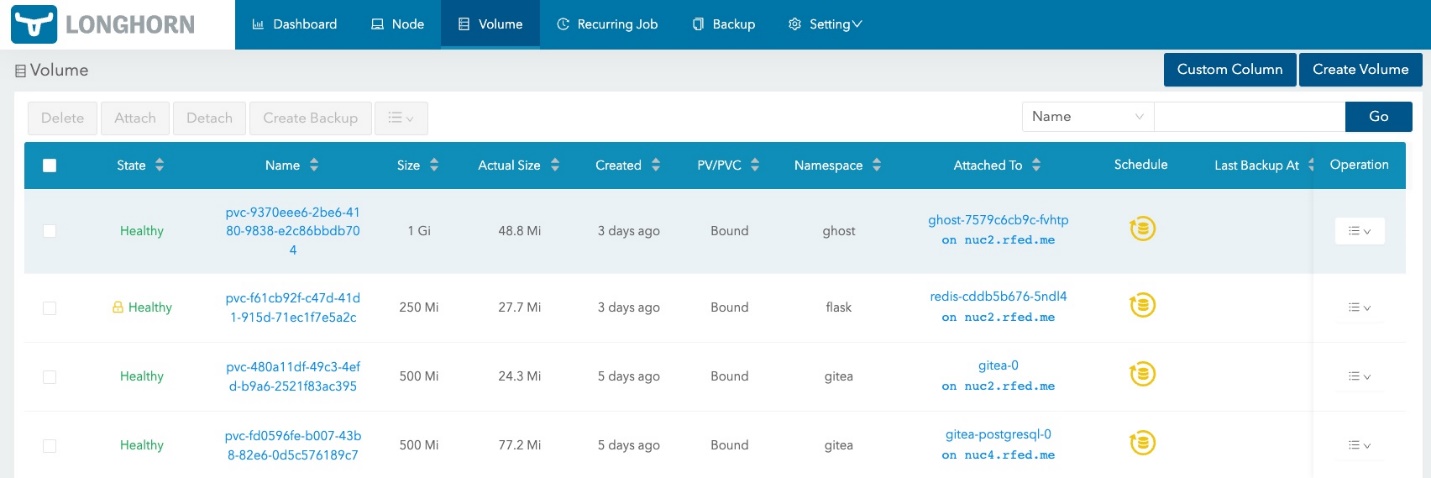
Since we leverage Kubernetes in our deployments, we use some of the tools in the Rancher portfolio, namely [RKE2](https://docs.rke2.io/) for the Kubernetes layer. Next, we use [Longhorn](https://longhorn.io/) for stateful storage across the nodes. And last but not least, we use the [Rancher](https://docs.ranchermanager.rancher.io/) Multi Cluster Manager to orchestrate everything. (More on RKE2, Longhorn and Rancher MCM in a moment). For all the installs we follow the airgap instructions, and for good reason. The downside of the tactical edge is that you must assume there is no network communication with the outside world.

### RKE2

What is RKE2 you ask? It is a fully conformant Kubernetes distribution that focuses on security and compliance within the U.S. Federal Government sector. Meaning it has FIPS, SELlinux, STIGs, compliance, and security support at its foundation. Another great reason to choose RKE2 for your Kubernetes layers is that air-gapping the software is not an afterthought. In fact, all of Rancher's products have airgap install instructions. For the sake of this guide, we will skip providing the code. Please review the airgap install [docs for RKE2](https://docs.rke2.io/install/airgap/). The basic procedure for installing air-gapped is to get the tarball that contains all the bits. It is also worth mentioning we have another article on applying the STIG and security best practices for RKE2 and Rancher on [Intelligence Community News](https://intelligencecommunitynews.com/ic-insiders-have-you-stigd-your-kubernetes-yet/).

### Longhorn

As mentioned above, Longhorn is Rancher's storage product. Longhorn creates a highly available, encrypted at rest if enabled, storage layer using the aggregate storage already on the nodes. It’s a fantastic way to create storage for stateful applications without having to add additional hardware.



Similar to RKE2, Longhorn has [very good documentation](https://longhorn.io/docs/1.3.1/advanced-resources/deploy/airgap/) for installing across the airgap. Longhorn and Rancher MCM (more below) use a similar model of moving container images and Helm charts. For this reason, it often makes sense to stand up a registry inside the kit.

### Rancher

Now let's look at our flagship product the Rancher Multi Cluster Manager. The Rancher MCM enables you to create a single pane of glass to manage all the applications in your kit. Rancher manages the application life cycles through a variety of methods, including GitOps. Adding version control within the kit will help facilitate GitOps. Rancher's primary method for installing is Helm. In order to install, air gapped charts will also need to be moved across with the images. Just like in the other products, the [Rancher Air-Gapped install](https://docs.ranchermanager.rancher.io/pages-for-subheaders/air-gapped-helm-cli-install) docs are very detailed.

At this point in our guide, we have walked through an edge kit that is fairly complete. However, there are a few applications that we can add to improve the functionality. They include:

### Harvester: Harvester is Rancher’s hyperconverged solution. [Harvester](https://www.rancher.com/products/harvester) can serve Virtual Machines (VMs) out from one of the nodes. Being hyperconverged means Harvester can support VMs and Kubernetes applications from the same node. In fact, the cluster in the pictures above is running Harvester on the third node. Harvester gives the added ability to serve Windows VMs to the kit. Another great use of Harvester is to serve more infrastructure related applications to the kit, like DNS or version control. One fun fact about Harvester is that it uses Longhorn under the hood for storage. In certain applications, it makes sense to run Harvester on all three nodes and then use the VMs to carve out small compute envelopes. This is a good practice for different security domains.

### Gitea: [Gitea](https://gitea.io/en-us/) is a great solution for in kit version control. In practice, Gitea becomes the source of truth for how the applications are deployed – aka GitOps. One pro-tip is to use a Longhorn volume for Gitea for highly available, stateful, storage.

### KeyCloak: KeyCloak is an authentication application that can provide Two Factor Authentication alongside SAML2 and OIDC. Basically, KeyCloak will give the kit a greater level of user management to not only Rancher but also to the applications that will be deployed into the kit.

### Registry: Similar to Gitea as the source of truth for files. A registry is a good idea as a source of truth for images. [Harbor](https://goharbor.io/) is a really good choice for a registry. While the docs to not clearly call out an air-gapped install they do have a section to [download the Harbor Install](https://goharbor.io/docs/2.3.0/install-config/download-installer/). Another alternative to Harbor is the original [Docker Registry](https://hub.docker.com/_/registry).

Hopefully, you now have a clear definition of the tactical edge and good understanding of how to deploy and leverage Kubernetes at the Edge. This guide is meant as a framework for implementing a similar kit that fits the applications compute envelope. At Rancher Government Solutions, we like to say that our software meets you at the mission – and this is one example of how.

To learn more about how Rancher deploys at the edge and supports mission critical work for customers, visit [www.rancherfederal.com](http://www.rancherfederal.com)

###