Kubernetes at the edge: How strict confinement enables a secure IoT landscape

October 2022

Executive Summary

Edge computing has become a critical component of the modern technology landscape. From the factory floor to smart homes, signage, robotics, and autonomous driving, Internet of Things (IoT) devices bring compute power closer to where data is generated, enabling unprecedented levels of efficiency and automation.

But as IoT use cases and capabilities evolve, so do the risks.

In the past, there were far fewer devices deployed in the field, and they typically remained in a fixed state throughout their entire lifecycle. Today, organisations often rely on Kubernetes to manage millions of devices, all of which receive a regular stream of updates and communicate with external data sources. Each of these interactions represents a window of vulnerability that can potentially be exploited by bad actors to gain access to the device.

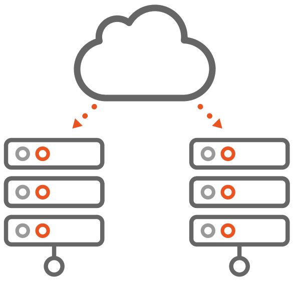
Given that even a single security breach at the edge can be highly costly and deal immense damage to an organisation's brand reputation, protecting IoT devices is paramount.

MicroK8s is a lightweight Kubernetes distribution designed for the edge that aims to eliminate these security concerns. Complementing a wide array of existing MicroK8s security features, a strict confinement capability introduced in 2022 provides complete isolation for containerised applications. This ensures that any breaches or vulnerabilities that occur on one container cannot spread to the rest of the system, drastically improving the overall security of the host IoT device.

Additionally, strict confinement enables users to run MicroK8s on Ubuntu Core, a Linux operating system purpose-built for secure edge computing.

This whitepaper will take a closer look at the security challenges that organisations face at the edge, and examine how those risks can be mitigated using Ubuntu Core and MicroK8s with strict confinement.

What is the IoT edge?



The Edge computing market is growing fast. According to a recent report by Gartner, more than 15 billion IoT devices will connect to enterprise infrastructure by 2029.[1] While traditional infrastructure remains relevant, the worldwide trend is to move resources closer to end users and IoT appliances.

That being said, the "edge" is a nebulous term with various definitions depending on the use case. The goal of all edges is largely common: providing compute, network, and storage resources closer to the source of the data. However, this definition captures everything from embedded-type single-board computers and IoT devices all the way to large point-of-presence (PoP) clusters of data centre class equipment. These use cases have vastly different requirements and constraints, making "edge" unhelpfully vague as a term.

This can be simplified by grouping edge computing into two categories:

- Micro clouds: small clusters of compute nodes with local storage and networking, where traditional data centre primitives are combined with high-availability clustering, low latency, over-the-air (OTA) updates, and enhanced security.

- IoT edge: Single node devices with fewer resources, which function more like appliances.

IoT edge devices are exponentially more prevalent, and this category is the primary subject of this whitepaper.

Edge security challenges

Keeping edge computing secure is a constant struggle because, in many ways, the same things that make modern IoT devices valuable also make them vulnerable.

Historically, IoT devices deployed in the field were immutable. After leaving the factory, the device firmware and operating system would never change, making security relatively straightforward. Modern IoT devices fall at the opposite end of the spectrum. Containerised applications are constantly updated, and advanced processes often interact with sensors and systems outside of the host device.

The ability to keep devices updated after deployment and connect them to external data sources is one of the primary factors behind the successful evolution of edge computing. But at the same time, exposing IoT devices to the world in this way dramatically multiplies the risks. Every update and external interaction is a potential vulnerability, and edge networks are notoriously complex to secure.

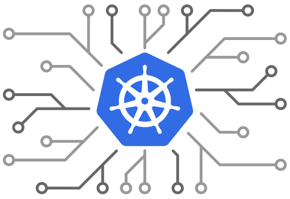
Maintaining security and delivering a consistent user experience in a dynamic edge environment is also a difficult balance to achieve. Developers might initially design a smart set of constraints for their IoT software, but those constraints can quickly become outdated as the landscape shifts and the device is patched.

[1] https://www.gartner.com/smarterwithgartner/gartner-predicts-the-future-of-cloud-and-edge-infrastructure

What's more, delivering updates to a fleet of devices is easier said than done. Automated OTA updates go a long way towards simplifying the process, but it is rarely possible to have full control over devices in the field. For instance, a user might have their device turned off for several weeks, preventing it from receiving updates. When that machine is turned back on, there may be a brief window in which it is operating without the latest security fixes.

These issues are further compounded with the introduction of Kubernetes.

Kubernetes complexity increases at the edge



Kubernetes is an open source container orchestration system. It is invaluable for distributing process management across numerous machines, enabling users to automate deployment, upgrades, and maintenance of containerised applications at scale.

Within data centres and on clouds, Kubernetes has been the industry-standard container orchestration platform for several years. Indeed, the latest Canonical Cloud Native Operations survey found that 43.1% of respondents ran applications either partially or exclusively on Kubernetes, with a further 29.9% evaluating or planning Kubernetes deployment.[2]

More recently, the quest to bring computing closer to users and data sources has seen Kubernetes come to the edge as well. According to the Cloud Native Computing Foundation's 2021 survey, edge developers are the largest user segment for both containers and Kubernetes, with Kubernetes used by 63% of developers working on edge computing applications in the last 12 months.[3]

Kubernetes is a natural fit for IoT since it offers a way to simplify deployment and management across a vast fleet of devices, empowering developers to focus on their applications rather than the underlying infrastructure. That being said, when Kubernetes was created, it was not designed to run at the edge. As a result, using Kubernetes on an IoT device poses significant risks.

Kubernetes is a highly dynamic platform that supports many applications across a wide variety of use cases. Often, these applications interact with host machines in ways that are not fully secure when transposed to the edge. For instance, Kubernetes makes certain assumptions about what it can touch on the file system, and how it can interact with network interface and storage devices. While these assumptions would not ordinarily be an issue, they can compromise security in an IoT context.

Combining these complications with device mutability and the sheer scope of what Kubernetes enables at the edge creates a large attack surface that is almost impossible to fully police.

[2] https://juju.is/cloud-native-kubernetes-usage-report-2022#container-and-kubernetes-usage

[3] https://www.cncf.io/wp-content/uploads/2021/12/Q1-2021-State-of-Cloud-Native-development-FINAL.pdf

Growing attention from bad actors

To make matters worse, the number and severity of security threats at the edge are growing rapidly.

In the past, malicious actors have focused more on finding and exploiting vulnerabilities on servers, laptops, and workstations than on IoT devices. But with embedded devices becoming more important, prolific, and mass produced, security researchers and hackers are increasingly turning their attention to the edge.

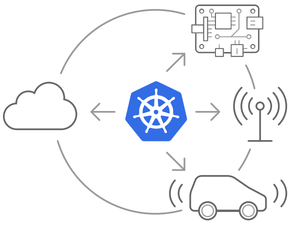
Cybersecurity leader Kaspersky reports that its honeypots – imitating vulnerable IoT devices – were attacked approximately 1.5 billion times in the first six months of 2021. That figure is more than double the 639 million attacks seen in the back half of 2020.[4] As edge computing continues to mature, the rise in vulnerabilities and hacking attempts will only accelerate.

Hardware integrity

While the security of IoT software is essential, it is also important to consider the integrity of the edge hardware itself. As devices trend towards increased computing power and reduced energy consumption, hardware security does not always keep up, adding a further layer of risk.

Unlike data centre hardware, IoT devices in the field can be physically accessed by malicious actors. As such, everything from low-level mainboard components to debugging interfaces and buses are viable targets, creating an assortment of new security challenges.

Solution: strict confinement with MicroK8s and Ubuntu Core



Despite all these challenges, there are solutions available that minimise the risks involved in running Kubernetes at the edge.

What is MicroK8s?

MicroK8s is a lightweight, CNCF-certified, pure upstream Kubernetes distribution designed not only for clouds and workstations, but also for IoT devices. Created by Canonical, the company behind Ubuntu, MicroK8s is delivered as a snap – a containerised software package that bundles Kubernetes together with all of its dependencies.

Optimised for security, simplicity, and robustness, MicroK8s can be deployed anywhere with a single command, offering the easiest path to enjoying the full Kubernetes experience at the edge. Additionally, the low footprint of MicroK8s makes it ideal for smaller, resource-constrained IoT devices. The distribution's low touch, minimal ops design aligns with IoT use cases where direct human intervention with devices in the field is impossible.

From a security perspective, MicroK8s includes self-healing, high-availability, and automated OTA updates. These updates are fully transactional and roll back on failure.

[4] https://www.iottechnews.com/news/2021/sep/07/kaspersky-attacks-on-iot-devices-double-in-a-year/

Most importantly, being a snap, MicroK8s is isolated from underlying systems, limiting its access to other system services and resources on an IoT device. To truly address edge security concerns, MicroK8s version 1.25 takes this concept a step further with strict confinement.

What are snaps?

Snaps are a secure and scalable way to embed applications on Linux devices. A snap is a software package containing one or more applications or services that are containerised with all their dependencies, which can be installed using a single command. With snaps, software updates are automatic and resilient. Applications run fully isolated in their own sandbox, thus minimising security risks.

Snaps are hosted in the global Snap Store, an application repository hosted and managed by Canonical, and are free for anyone to download. While strict confinement requires the use of Ubuntu, snaps can be installed on any Linux distribution with snap support enabled.

What is strict confinement?

Strict confinement is a snap confinement level that provides complete isolation up to a minimal access level that is always deemed safe. With strict confinement enabled, the system ensures that MicroK8s and its container workloads can only access files, system resources, and hardware for which access has been granted.

By restricting Kubernetes to the absolutely necessary permissions, strict confinement eliminates vulnerable interactions both within the host device and externally, greatly reducing the attack surface. And if MicroK8s were to be hacked, strict confinement would prevent it from compromising the rest of the device.

With strict confinement, users can run sophisticated and otherwise high-risk IoT workloads in a safe way. The ability to layer applications together in a hardened environment without risk of device-wide intrusion enables a variety of unprecedented use cases, empowering businesses to operate at the edge in entirely new ways.

How does strict confinement work?

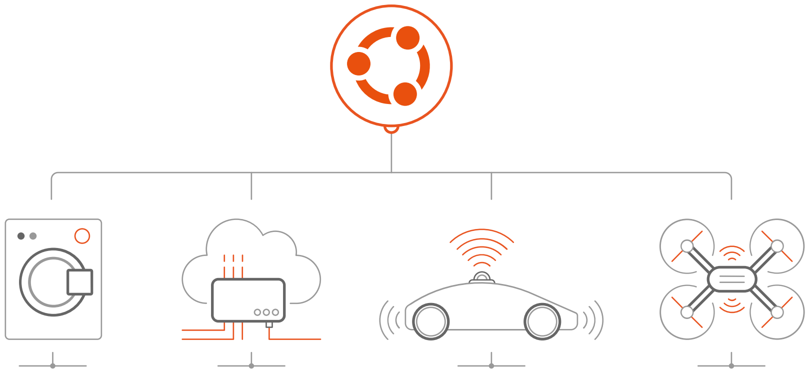
To determine their permissions, strictly confined snaps rely on resource access requests known as interfaces. Interfaces are carefully chosen by a snap's creator to provide specific access to a resource according to that snap's requirements. For instance, interfaces can provide access to cameras or serial ports.

Confinement and permissions are enforced by the Linux kernel security module AppArmor, alongside other Linux security features. When strictly confined MicroK8s is installed, its metadata is examined and used to derive AppArmor profiles, seccomp filters, device cgroup rules, and traditional permissions. Together, these ensure total isolation for the Kubernetes runtime.

Naturally, some applications require access to critical system resources in order to function, and so need to be exempt from confinement. To support these use cases, MicroK8s features an addon system with verified and tested applications that will run under strict confinement. Canonical is working to continuously improve this ecosystem to provide support for all common use cases.

Perhaps the most important consequence of strict confinement being added to MicroK8s is that it can now be used with Ubuntu Core.

What is Ubuntu Core?



Ubuntu Core is a lean, embedded version of Ubuntu created for the edge. The main goal of Ubuntu Core is to secure the next generation of IoT devices, and it achieves this through containerisation. Ubuntu Core itself and all applications deployed on it are packaged as strictly confined snaps.

This snap-based paradigm takes the benefits of strict confinement detailed above and proliferates them throughout the entire device. All applications are fully isolated from each other and can only interact with the system through interfaces. This approach is inherently secure and perfect for IoT devices.

Alongside application confinement, Ubuntu Core features transactional OTA updates with self-healing, full disk encryption, secure boot, and an array of additional capabilities that make the operating system ultra-secure straight out of the box. Canonical supports Ubuntu Core long-term, delivering kernel patches and bug fixes continuously for 10 years. Each Ubuntu Core version is based on a corresponding LTS release of Ubuntu.

Because all applications running on Ubuntu Core must be strictly confined, it was not previously possible to pair it with MicroK8s. Now, Ubuntu Core and MicroK8s can be combined for a seamless path to secure Kubernetes at scale, optimised for size, performance, and usability at the edge.

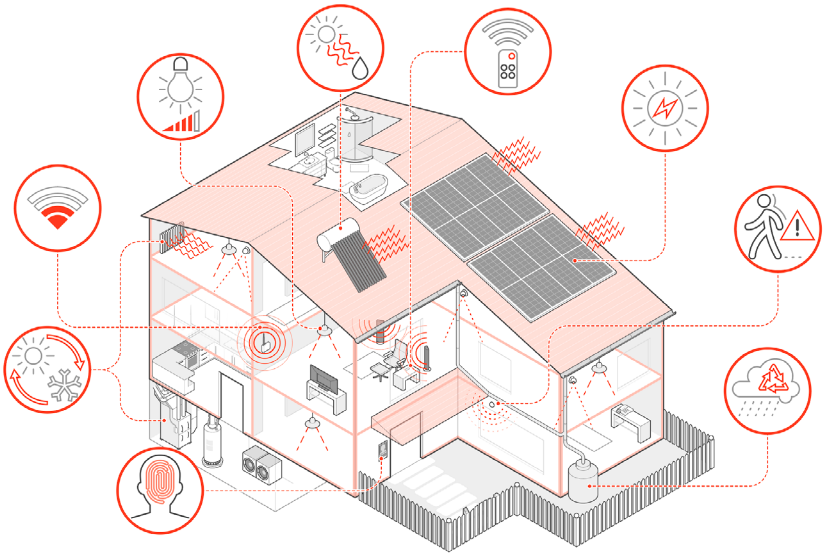
Use cases for strict confinement

We have discussed the technologies, but how do they work in practice? Let’s explore the use cases.

Smart home

Let’s examine a hypothetical application of MicroK8s in a smart home. In this smart home, the fridge acts as a hub controlling multiple smart devices in the house. That hub is running Ubuntu Core with MicroK8s.

The workloads on the hub communicate with sensors and other devices around the house. Critically, these devices must not be influenced by outside updates that could cause a malign threat later down the line. All updates must come from a secure source.

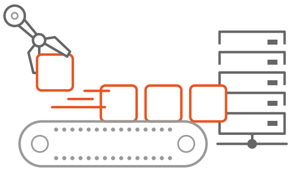


One day, an update provides a new type of workload with a new MicroK8s pod that connects to the TV. It locally stores the latest data coming from the TV on the fridge database, on the Ubuntu Core file system.

With strict confinement, if the Microk8s pod ever encounters unvalidated or unsafe data, it can still store it without risk. If it is malware, it cannot reach outside of its MicroK8s sandbox or compromise any other part of the system.

In this way, strictly confined MicroK8s lets device manufacturers and smart home platform providers expand their offerings and deliver the latest feature updates with confidence that they will not be putting their customers' homes and data at risk.

Smart factory



Edge computing is central to the Industry 4.0 manufacturing revolution. Interconnected IoT devices capture, share, and process data directly on the factory floor, enabling smart factories to achieve a significant degree of autonomy.

Typically, industrial IoT devices have been limited to a single function, such as a programmable logic controller (PLC) or an industrial gateway. Snaps enable manufacturers to take the next step and split IoT devices into different containers, with multiple functions each sitting in their own snap.

Strict confinement keeps this snap-enabled, software-defined strategy secure by ensuring that the various functions on an industrial IoT device will only access the appropriate, permitted resources and services. For example, take a user who wants to run AI/ML workloads at the edge to predict machine failure based on vibrations and enable proactive maintenance before it happens. By using Ubuntu Core, the user can run these workloads in a snap on the same software-defined device that is already in use for monitoring the environment. For instance, the device may further measure ambient temperature, luminosity and humidity levels on the factory floor and share those with the IT department.

Without strict confinement, the sensors' measurements may be exposed to the AI/ML algorithm. This would result in unbalanced datasets leading to biased predictions that could ultimately lead to machine downtime. With strictly confined MicroK8s, the AI/ML app would only "see" what is in its predefined, sandboxed environment, eliminating this risk.

Conclusion

Strictly confined MicroK8s on Ubuntu Core is the easiest and safest way to utilise Kubernetes at the edge. Being deployed outside the data centre and often beyond the reach of human intervention, IoT devices will never be free from security risks – but strict confinement and snap architecture ensure that vulnerabilities remain isolated to a single application, with seamless remote resolution.

Resources

- For a tutorial on getting started with MicroK8s on Ubuntu Core, visit: ubuntu.com/tutorials/getting-started-with-microk8s-on-ubuntu-core#1-introduction
- To learn more about strictly confined MicroK8s, including a live demo, watch the webinar: ubuntu.com/engage/embedded-kubernetes-for-secure-iot-edge-webinar
- For guidance on implementing a successful edge strategy, read the whitepaper: ubuntu.com/engage/edge-infrastructure
- Contact Canonical to learn about enterprise MicroK8s support: microk8s.io/contact-us

Canonical Limited, Registered in Isle of Man, Company number 110334C, Registered Office: 2nd Floor, Clarendon House, Victoria Street, Douglas IM1 2LN, Isle of Man, VAT Registration: GB 003 2322 47

Ubuntu