KUBERNETES SECURITY GUIDE

TechRepublic.

KUBERNETES ROLLOUTS: 5 SECURITY BEST PRACTICES

If you don't follow these Kubernetes deployments security best practices from Portshift, your containers, their underlying technologies, and your data could be at risk.

BY JACK WALLEN

Within the realm of Kubernetes (K8s) security, Portshift is one of the many industry leaders; the company focuses on identity-based workload protection for cloud-native applications. Portshift offers solutions for Kubernetes, Zero Trust security, DevOps, and compliance.

IMAGE: ISTOCKPHOTO/123DARTIST

Anyone that has followed Kubernetes and containers over the past few years knows that security has become a central point of failure for this technology. Security issues can arise from nearly any point—from container images, runtime engines, poorly secured networks, etc. So for any business looking to adopt container technology, the importance of security cannot be overstated.

Portshift recently released a best practices list for tackling the security issues surrounding the K8s platform. Let's look at these security tips.

1. AUTHORIZATION

Authorization is probably one of the most overlooked issues with Kubernetes. Why? It's not always a simple hurdle to overcome, because you have to deal with authorization on multiple levels. Consider this: You have

authorization from within images, configuration files, third-party applications and services, various developers and/or users… the list of possible authorizations goes on and on. That is why permissions in Kubernetes is handled by role-based access control (RBAC).

This mechanism gives you powerful, fine-grained control over authorization and access. The RBAC API declares four top-level types:

- **Role** can only be used to grant access to resources within a single namespace;
- **ClusterRole** adds cluster-scoped resources, non-resource endpoints, and namespaced resources to the Role type;
- **RoleBinding** grants permissions defined in a role to a user or set of users and
- **ClusterRoleBinding** is the same as RoleBinding, but across a cluster.

It is imperative that every K8s admin understand RBAC authorization. For more information, make sure to read the official RBAC documentation.

The Portshift best practices list also includes the ABAC authorization method, but warns that it does include a few operational constraints.

2. POD SECURITY POLICIES

The next best practice is pod security. A pod is an object that contains a set of one or more containers. According to Portshift's best practices, "it is essential to control their deployment configurations. Kubernetes Pod Security Policies are cluster-level resources that allow users to deploy their pods securely by controlling their privileges, volumes access, and classical Linux security options such as seccomp and SELinux profiles."

Note: A Pod Security Policy controls sensitive aspects of your pod specification. The PodSecurityPolicy object defines a set of conditions a pod must achieve in order to be accepted into the system; if the pod cannot achieve such a state, it will not be accepted. Pod Security Policies allow an admin to control such things as:

- Running of privileged containers
- Usage of host namespaces
- Volume type usage
- Host filesystem usage
- Requiring usage of read-only root file system
- User and group container IDs
- Restrict escalation to root privileges

These are fairly sensitive aspects to your pods, and you need to pay close attention to not only how you set your Pod Security Policies, but who has access to them.

3. SECURE THE PRODUCTION ENVIRONMENT

The security of your Kubernetes deployment is only as sound as the production environment it is deployed from and to; this should go without saying, but it does get overlooked. Portshift says this about the issue:

"As companies move more deployments into production, that migration increases the volume of vulnerable workloads at runtime. This issue can be overcome by applying the solutions described above, as well as making sure that your organization maintains a healthy DevOps/DevSecOps culture."

Your production environment must be secure. From your networks to your development environment, including developer desktops, servers, and--as Portshift pointed out--your DevOps culture. If your developers aren't working in a secure environment, the chances increase that your Kubernetes deployments can be compromised.

4. SECURING CI/CD PIPELINES

Continuous Integration/Continuous Delivery (CI/CD) allows you to do pre-deployment build-outs, testing, and deployment of workloads; it also enables the automation of many deployment tasks. To make this work, you'll use a number of third-party tools such as Helm and Flagger.

In order for your Kubernetes deployments to enjoy even a modicum of security, you must lock down everything within your CI/CD pipelines. You absolutely must roll in tight security practices into this pipeline and every piece of software or service that touches it; otherwise, according to Portshift, "attackers can gain access when these images are deployed and exploit these vulnerabilities in K8 production environments. Inspecting the code of images and deployment configurations at the CI/CD stage can achieve this purpose."

This particular portion of Kubernetes is where a lot of security can break down. If you're unsure of how a particular tool accesses your CI/CD pipeline and how it handles things like authorization, learn everything you can about it. A single point of failure in your CI/CD pipeline could be catastrophic to the security of your Kubernetes deployments as a whole.

5. ADD SERVICE MESH TO THE NETWORK SECURITY LAYER

Network security is crucial to your Kubernetes deployments and should not be overlooked. A service mesh boosts network security by adding a dedicated infrastructure layer to facilitate service-to-service

communications between microservices and balances inter-service traffic based on specific policies.

To this issue, Portshift says:

"It [service mesh] also offers a number of security, reliability, and observability benefits that can help manage cluster traffic and increase network stability that is enhanced by a 'zero-trust' security model."

Istio is currently your best bet for service mesh. Istio helps you to intelligently control the flow of traffic and API calls between services, automatically secure your services through managed authorization, apply policies, and observe what is happening with automatic tracing, monitoring, and logging.

This is yet another layer you are adding to an already complicated fabric of layers, so before you employ the likes of Istio, be sure you understand it.

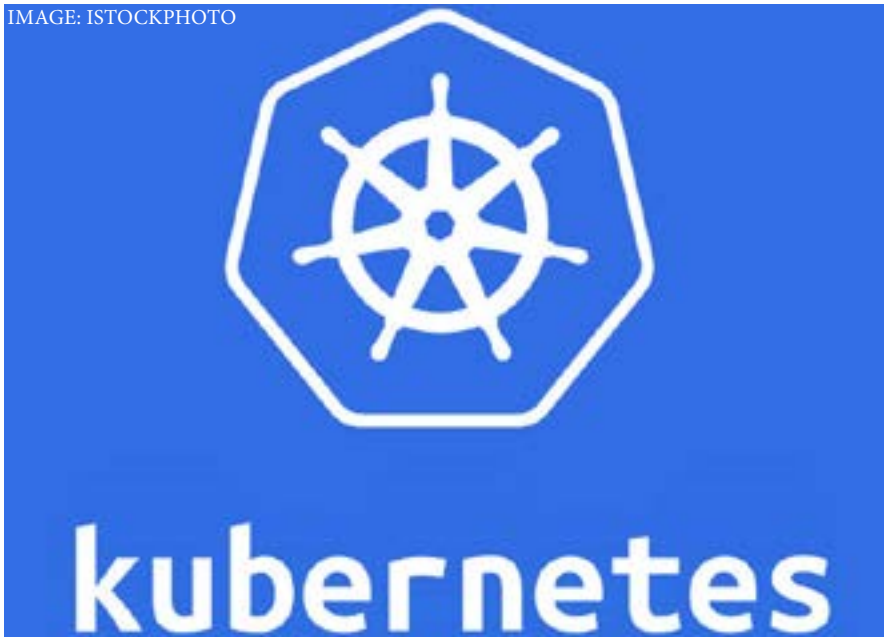
3 TIPS TO KEEP KUBERNETES SAFE AT SCALE

As more companies adopt and scale their Kubernetes systems, security has to be a major point of interest.

BY JONATHAN GREIG

Kubernetes containers are now highly prevalent in multi-cloud environments and are being deployed widely across a variety of industries. In a survey last year, vice president of product marketing for Sumo Logic Kalyan Ramanathan wrote that the open-source container operating system was "dramatically reshaping the future of the modern application stack."

IMAGE: ISTOCKPHOTO



KubeCon 2019, a Kubernetes conference in November 2019, drew more than 12,000 people, and about 400 attendees reported that their company planned to run at least 50 clusters in production in the next six months.

But with the expansion of adoption have come security concerns, which reared their heads throughout 2019 in a series of security flaw vulnerabilities discoveries. The first was found at the end of 2018 and involved a privilege escalation flaw that allowed any user to establish a connection through the Kubernetes application programming interface server to a backend server.

Many companies, like Capital One and Walmart, used the conference to advertise their goals and meet with experts while hunting for Kubernetes-trained talent. Kamesh Pemmaraju, head of product marketing a Platform9, said the number of clusters and nodes being used were increasing, with some respondents telling Platform9 that their company was running hundreds of nodes in only one or two clusters. The survey also highlighted the fact that many companies are running Kubernetes on both on-premises and public cloud infrastructure.

NeuVector CTO Gary Duan and Platform9's co-founder and CTO Roopak Parikh spoke to TechRepublic about three ways to leverage Kubernetes' security capabilities at scale.

"Kubernetes has some basic security features--primarily around securing its own infrastructure. These include role-based access controls (RBACs), secrets management, and pod security policies (which are more focused on resource management)," Duan said.

"However, Kubernetes should not be deployed in business-critical environments or for applications that manage sensitive data without dedicated container security tools. As we've seen in the past year, the Kubernetes system's containers themselves--such as the critical api server--can have vulnerabilities that enable an attacker to bring down the orchestration infrastructure itself," he added.

1. ROLE-BASED ACCESS CONTROL

Parikh said his team at Platform9 is seeing a lot more production applications than last year and a lot more data or other applications being deployed on Kubernetes. Companies should look at Kubernetes as a complex distributed system made up of multiple components that need every layer secured when put into production.

"At the application level, we highly recommend putting in network policies and connecting Kubernetes clusters to an authentication provider that's within the enterprise, so at a directory or one login. A lot of times what we have seen is, you can secure all your systems all you want, but it's also important to secure the keys, as in the passwords you use, with multi-factor authentication and use certificates to make sure people cannot get in through those easy doors," Parikh said in an interview.

"As always, security is about multiple layers. There is role-based access control, network policies and a few others. When your applications are running, how and where are you storing some of the secrets? If your application needs access to a database, the application needs to connect to the database using some credentials for where you are storing it."

Kubernetes operators should make sure users have the correct roles and assign users to different spaces so that specific users are associated with specific applications. These roles would also be scalable as systems are deployed and upgraded, Parikh added.

In general, each application in a cluster should be segregated or isolated so that the person in charge can decide what users can see different parts of the system.

Duan noted that a layered security approach was best for in-depth defense. Cloud-native security tools should be deployed to secure the entire lifecycle, from the CI/CD pipeline to run-time, because traditional security tools don't work in Kubernetes environments, Duan said.

"This starts with vulnerability scanning during build and in registries, and then carries on into production. True defense in depth is not possible without deep network visibility and protection, such as with a Layer 7 container firewall. Such a container-focused firewall will be able to detect and prevent network based attacks, probes, scanning, breakouts, and lateral movement between containers using container micro-segmentation techniques," Duan said, adding that as companies expand their deployments, they face the management headache of managing and securing dozens or even hundreds or separate Kubernetes cluster, with some across public and private clouds.

Having global multi-cluster management with federated global security policies enforced centrally becomes a critical issue, he said.

"Layering a service mesh such as Istio or Linkerd on top of Kubernetes can also improve security by encrypting pod-to-pod communications. Service meshes have other benefits that DevOps teams are excited about. But again, this introduces additional attack surfaces into the infrastructure which must be secured," Duan said.

2. INSECURE CODE CHECKS

Parikh said monitoring code was vital to keeping Kubernetes secure at scale. If your company is running third-party applications or even internal applications, how do you know that the code that's there is secure or not? System operators should ask themselves what version they are running and what kind of security vulnerabilities are there.

"You may want to scan it to figure out if there is a insecure code or if you're using an older version of Python library. There are tools out there which can help you find out those details and give you a report on that. Based upon that, you can either deny or allow running those applications in the form of containment," Parikh noted.

Duan echoed those comments, adding that ultimately Kubernetes provides the mechanisms for automation that should be leveraged by security tools.

Companies have the ability to declare security policy as code, where the application behavior of new services being deployed is captured in a standard yaml file and deployed natively by Kubernetes as a custom resource definition. Security teams also have to make sure all of the patches are up to date.

"With the server and operating system that you're running, you need to make sure it's patched and that it doesn't have any vulnerabilities associated with the operating system itself. You're using technology like App Armor or to making sure you're giving the least privileges to the components that are running outside the

Kubernetes as well as on the Kubernetes itself," Parikh said.

"You need to make sure that every component you install is patched and up to date."

3. CHECKING EXPOSED PORTS

Duan said it is critical to define all the integration points in the pipeline where security should be automated and then build a roadmap for complete security automation. Initially, a few steps may be automated, such as triggering vulnerability scans and alerting on suspicious network activity.

According to Parikh, security teams need to protect applications that can connect to the outside world using load balancers or ports that can be opened up on your Kubernetes cluster themselves. Teams need to check whether they are configured correctly and if they are exposing it through the right security grids.

This is especially key with API servers to the external world. Parikh said in the logs of Platform9's customers they have seen a lot of services trying to figure out if they are running PHP.

"Someone figured out that something was exposed to the internet by other companies and we saw in the logs some people trying to probe our servers to figure out if those ports were open because insecure ports were open. Kubernetes itself has an API server, and in the past we have seen exposed ports," Parikh said.

"Are you securing your host policies to make sure you have the correct firewall running? If you're running in a public cloud, make sure you're using the correct security group and allowing only the actual action you need to. Are you making sure your Kubernetes components have the right kind of authentication or authorization so that only some users are able to login?" Security concerns hampering adoption of containers and Kubernetes Security concerns hampering adoption of containers and Kubernetes

SECURITY CONCERNS HAMPERING ADOPTION OF CONTAINERS AND KUBERNETES

According to a StackRox study, more than 90% of respondents have experienced a security incident in deployments in the last year.

BY JONATHAN GREIG

Enterprises are having significant problems with security when it comes to Kubernetes and container deployments, according to a new survey from security company StackRox.

IMAGE: ISTOCKPHOTO/ NAT'TAPON KONGBUNMEE

In the winter 2020 edition of its State of Container and Kubernetes Security Report, StackRox researchers found that 94% of respondents experienced a security incident in their Kubernetes and container environments during the last 12 months.

This very high number of security incidents led to about 44% of organizations delaying or outright halting application deployment into production.

Researchers spoke with more than 540 IT professionals, the majority of whom work for tech companies or organizations involved in financial services.

"Our survey data affirms what we hear anecdotally from customers, that security has become a high priority as customers seek to deploy containers and Kubernetes applications in production," said Kamal Shah, CEO of StackRox.

"Organizations have executive buy in – the challenge is understanding the security and compliance requirements so that they can be addressed early in the application development life cycle and prevent delays to application deployment."

Over the last five years, companies have been eager to incorporate containers, Kubernetes and microservices applications in an effort to promote enterprise IT innovation and boost digital transformation.

But nearly half of all respondents to the survey have had to delay an application rollout due to concerns about the security of containers or Kubernetes.

Of the 94% of respondents that acknowledged having security incidents, 69% said they experienced a misconfiguration incident and another 27% said they had a security incident during runtime. Nearly 25% reported having had a major vulnerability to remediate.

Exposures due to misconfigurations were considered the most pertinent security risk for their container and Kubernetes environments. More than 60% of respondents cited this as their main concern with another 27% telling StackRox researchers that vulnerabilities were also a big problem.

Misconfiguration incidents were particularly high because of how challenging it can be to find the kind of tech talent that can deal with the intricate knobs and dials in containers and Kubernetes.

Even seasoned developers have difficulties managing containers and Kubernetes, and many data breaches and exposures are caused by human error, the report said. More than 20% of enterprises experienced two or more types of security incidents.

"Companies understand they can't realize the advantages of containers and Kubernetes without getting security right. To see such a large percentage – 44% – acknowledge they've slowed or halted application deployment into production due to security concerns means these companies are not achieving the primary advantage – faster app delivery – of moving to containers," the report said.

"The findings in this survey of 541 respondents make clear that organizations are putting at risk the core benefit of faster application development and release by not ensuring their cloud-native assets are built, deployed, and running securely".

"With the prevalence of misconfigurations across organizations, security must shift left to be embedded into DevOps workflows instead of 'bolted on' when the application is about to be deployed into production," the report said. "With nearly half of our respondents delaying going into production because of security concerns, clearly a lack of security is inhibiting business acceleration and innovation."

The study said that for the third time in a row, lackluster investment in security was the most common concern IT departments have about the strategy used for containers. Nearly 40% of respondents said inadequate investment was their main concern while another 14% said their organizations did not take threats to containers seriously. More than half of all respondents said security was their biggest source of concern.

In a sign that this problem was being addressed, StackRox researchers said there was a 35% drop in respondents saying their security strategy isn't detailed enough. According to the study, the number of respondents with an intermediate or advanced strategy grew to 48% from 41%.

The survey also found that the most popular architectural model for deploying containers was hybrid because 46% of respondents said they were running it both on premises and in the cloud. More than 50% of survey respondents said they ran their containers on a single cloud platform while 35% ran theirs on multiple public clouds.

A RACE FOR THE TOP

The leading container providers are Amazon Web Services, Microsoft Azure and Google Cloud Platform. Amazon is far outpacing the competition, with 78% of respondents saying they use AWS and just 39% deploying Microsoft's. But the race between Microsoft and Google for second place is heating up.

"While Microsoft Azure remains in second place, Google Cloud Platform (GCP) has grown its third-place standing from 28% in spring 2019 to 35% today. That GCP rivals Azure so closely might not be surprising, since Google created Kubernetes before donating it to the Cloud Native Compute Foundation.

Also, Google Kubernetes Engine is one of the most feature-rich managed Kubernetes services in the market, especially in the area of cluster management – again, in large part due to Google's deep experience orchestrating containers at scale," the report said.

Kubernetes is also a dominating force in the market according to the survey's findings. Nearly 90% of respondents are using Kubernetes for container orchestration, but the skills shortage is hampering the ability of organizations to fully deploy the environments.

DevOps departments were cited as the main group put in charge of managing container security at 81%, with another 51% reporting that security teams were also seen as most responsible for keeping containers safe. But the report said security required coordination between security departments, DevOps teams as well as developers.

"One of the most consistent results we get on our own surveys of DevOps and cloud-native security technologies is how important security is for those environments," said Fernando Montenegro, principal analyst on the information security team at 451 Research.

"It is interesting to see how this observation fits well with the StackRox study, highlighting the need for both engineering and security professionals to properly deploy security controls and practices for containers and Kubernetes environments."

CREDITS

Editor In Chief
Bill Detwiler

Editor In Chief, UK
Steve Ranger

Associate Managing Editors
Teena Maddox
Mary Weilage

Editor, Australia
Chris Duckett

Senior Writer
Veronica Combs

Editor
Melanie Wolkoff Wachsman

Associate Staff Writer
Macy Bayern

Multimedia Producer
Derek Poore

Staff Reporter
Karen Roby

Cover Photo
iStockphoto/ Oleg Mishutin

ABOUT TECHREPUBLIC

TechRepublic is a digital publication and online community that empowers the people of business and technology. It provides analysis, tips, best practices, and case studies aimed at helping leaders make better decisions about technology.

DISCLAIMER

The information contained herein has been obtained from sources believed to be reliable. CBS Interactive Inc. disclaims all warranties as to the accuracy, completeness, or adequacy of such information. CBS Interactive Inc. shall have no liability for errors, omissions, or inadequacies in the information contained herein or for the interpretations thereof. The reader assumes sole responsibility for the selection of these materials to achieve its intended results. The opinions expressed herein are subject to change without notice.