The Dark Side of Containers: Protecting Container Data from Itself

IF A CONTAINER CREATES PERSISTENT APPLICATION DATA, THEN IT NEEDS TO BACK UP THAT DATA TO PROTECT IT AGAINST DATA LOSS, ERRORS, AND CORRUPTION.

— Originally posted July 30, 2018, on DATAMATION.COM, by Christine Taylor

Containers are just virtual machines, right? A container is a stand-alone, executable software that shares operating systems and houses application code and data, runtime, toolsets, libraries, and configuration. Created initially for Linux environments, it is also available for Windows Server. Container applications are platform-agnostic, meaning that they run as well in test/dev environments as they do in application deployments. Admins can create, retire, or rebuild containers at will.

Containers are not precisely selling like hotcakes, but they're making inroads. Performance vendor NGINX recently surveyed 1,800 IT pros on their container use. Adoption rates in production environments are low: just 20% are currently using containers in production. However, interest is growing, with 2/3 of the respondents reporting that they are actively evaluating containers. Most of this interest is specific to open-source Docker technology, the leading developer in the container space.

However, there's an issue – what happens to container data? There are two types of data in containers: the container image and application data. The image is controlled by the Docker Engine that creates, retires, and rebuilds containers. Application data may also be ephemeral. For example, testing containers exist only until the tests are complete. Then developers retire the container.

But when a container holds persistent data such as business databases, that data must be protected.

BACKGROUND: HOW CONTAINERS WORK

Let's start with a brief run-down on how containers work, and why they are not the same as VMs. We'll use market-leading Docker technology for the discussion.

Deploy Docker Machine on a host, whether a VM, physical server, cloud, or laptop (containers are platform- and device-agnostic). Docker Machine provisions the host for Docker operations and deploys Docker Engine for the runtime environment. To create a container, use Docker's build process to create a Dockerfile. The file is an immutable Docker image. The container works by combining the image with a writable layer that stores runtime changes and, in some cases, application data. A run command launches the image onto the host.

Containers can run on clouds, but not all clouds support Docker containers. The Docker Cloud itself does not host containers, so container users must provide their hosts. The three hyperscaled megaclouds – AWS, Azure, and Google – support container computing services. The specialized data-protection KeepItSafe Cloud has extensive support for Docker backup and restore, as well.

Finally, many broadcasters believe that they can restore from archives. And so they can – if they're willing to wait extra hours or even days. Archiving is not backup.

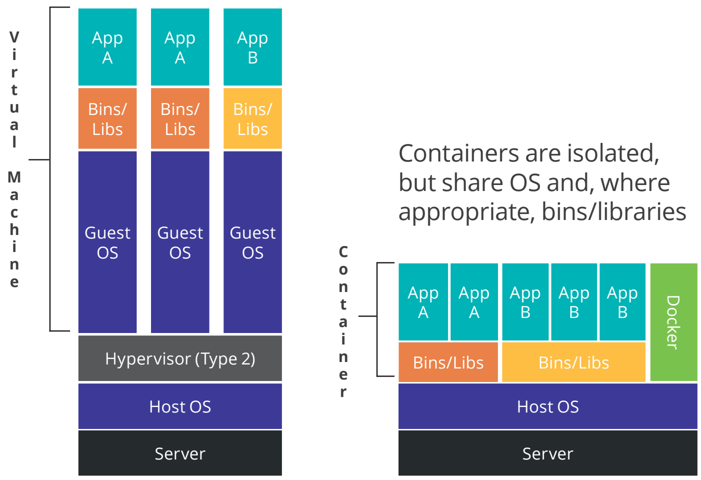
Containers and Virtualization

Containers are virtualized but not by hypervisors. They can be deployed to a VM but are not VMs.

Both containers and VMs use server/host OS as the bottom two layers of the stack. In VM environments, the next level is the hypervisor followed by VMs containing guest OS, libraries (div/lib in Linux), and applications. A single VM runs two full operating systems: the host and guest OS.

In contrast, containers do not have a hypervisor layer. A container shares the host OS, housing only the libraries and application code and data. Container benefits include greater portability, less operational overhead, lower OS licensing and maintenance/support costs, and less expensive application development.

Containers vs. Virtual Machines



Containers are isolated, but share OS and, where appropriate, bins/libraries

Orchestration tools and application support services expand container functionality.

Orchestration platforms manage multiple containers with scheduling, cluster management, and resource provisioning. The most widely used orchestration platform is Google-developed Kubernetes, open source since 2014. This toolset enables large numbers of containers to co-exist peacefully without a large operational investment. The orchestration platform allows container owners to:

- Run containers across different computing devices.
- Easily create, move, deploy, and retire containers.
- Keep data storage consistent across multiple application instances.
- Balance workloads between containers.

Google Cloud closely integrates Kubernetes with Google Container Engine for cloud-based containers.

Microsoft centralizes its container technologies on Azure with Azure Kubernetes Service, which replaced Azure Container Service. AWS supports containers with Amazon Elastic Container Service (ECS). Docker makes its own orchestration platform, Docker Swarm. Apache Mesos is a large-scale container cluster technology popular with container uses like Twitter and Apple Siri.

Application support tools improve container application performance. For example, many of these toolsets improve container application networking. Docker Networking virtually connects containers, while open source Project Calico uses IP routing to connect containers with VMs and distribute network security policies.

THE $64,000 QUESTION: DO YOU HAVE TO BACK UP CONTAINERS?

Not every container needs to be backed up. Container data consists of two types of data: the application code/image, such as a database, and the application data (if any).

There is no need to back up a container image; it can simply be restarted. When a container launches, the runtime engine checks for a local instance first. If it doesn't find it – let's say the container was accidentally deleted off the local machine – it simply recreates it. Moreover, if the container does not create application data, then there is no need to back up. These types of containers are ephemeral/temporary, create no data, and have identical configurations.

But if a container creates persistent application data, then IT needs to back up that data to protect it against data loss, errors, and corruption.

Backing up container data to off-site cloud storage, as opposed to transient hosts, makes for the best data-persistence in containers.

- **Reliability**. Backup software automatically validates container data backup in the background, which ensures full recoverability.
- **Manageability**. A central management console allows IT to treat container backup as part of a holistic backup strategy.
- **Flexibility**. Look for container backup that can use policies and automated operations. This gives you the flexibility to assign different backup schedules and storage targets to container data.
- **Fair pricing**. Understand the backup application's pricing structure, including purchase, licensing, maintenance and support, and minimum hardware requirements. If you go with a cloud subscription model, understand how the provider calculates monthly fees.

Types of Container Backup

There are several ways of doing container data-protection, including mounting, plug-ins, traditional backup applications, volumes, and scripting. They are not necessarily all good ways, but ways they are. Here are a few of the more common methods:

- Follow the 3-2-1 rule. This older rule still applies in modern backup strategies.
    - Maintain at least 3 copies of back up data – active data and 2 backups.
    - Store your backups on at least 2 different media types. It's unlikely that nearline disk and off-site tape or cloud will go down at the same time.
    - Keep 1 of the backups in a different location. If all your backups are stored in the data center, then good luck to you should the data center go down.
- For containers running database applications, store the database outside of the container on the same server/host. Direct the application's writes and reads to the database and use the database backup software as you normally do.
- Store your database in a Docker Volume, which enables snapshots and replication. Be wary, though; this method results in a crash-consistent backup and not an application-consistent one. For that you would have to quiesce the database to take the snapshot.
- Deploy traditional backup software into the container and back up application data to persistent storage. This method has the advantage of using the same backup software across the enterprise but can consume a lot of resources and compromise container efficiency.

CONTAINER BACKUP VENDORS

One backup offering that does not compromise data persistence or efficiency is a partnership: Asigra Cloud Backup with KeepItSafe Cloud.

Asigra, which first developed commercial Docker backup, is an exception to the heavy resource usage of other traditional backup applications. Asigra has supported Docker container backup since 2014. As of v14, Asigra added its agentless DS-Client that runs directly from a container and is available for download from Docker Hub.

DS-Client is an orchestration application. Deploying its container to an unprotected host that is running Docker Engine makes Asigra Cloud Backup immediately available. The software backs up only changed data in a compressed and deduped format, encrypts the data, and transports it to the storage platform. Admins can set policies and schedules for container backup, including continuous data protection for high-value data. Backup runs transparently in the background with minimal resource consumption – maximizing cloud consumption and making for a cost-efficient backup solution.

The software is tuned to back up Docker container application data to holistic, “as-a-Service” cloud providers like KeepItSafe, or to any public megacloud vendor.

The KeepItSafe custom-tuned cloud specializes in protecting highly regulated, long-term-retention, and high-value databases. This type of comprehensive data-availability solution improves data protection across enterprise data with simplified management, automated backup, verified backup, and rapid restore.

Other vendors also offer container backup. Blockbridge plug-in for Docker Volume backs up and restores container data. Users need to install Blockbridge Elastic Programmable Storage as the storage back end. Commvault backs up container images and data by adding a virtualization client to Docker hosts.

NetApp makes a plug-in that works on NetApp hardware and software to create container snapshots and clones. HPE Nimble Storage uses a container plug-in on its flash arrays, while Pure Storage offers Docker and Apache Mesos container backup on its storage systems.

Backing up containers doesn’t end with the backup vendor. You also need to trust the backup target as to its availability, durability, verification, cost-effectiveness, and whether is possesses a secure cloud environment for browser-based restore from anywhere.

Forward-looking organizations are adopting containers for efficient application development and deployment. These businesses understand that backup is critical to protecting enterprise data no matter where it resides.

About KeepItSafe

KeepItSafe provides global cloud data availability through its Backup-as-a-Service (BaaS), Disaster Recovery-as-a-Service (DRaaS), endpoint protection, and cloud SaaS application backup solutions. Backed by a $1.2 billion public company, j2 Global,® Inc. (NASDAQ: JCOM), KeepItSafe meets data-security protection regulations with ISO 27001, SOC 2, HIPAA, and PCI compliance in 20+ data centers across three continents. KeepItSafe’s holistic approach leverages its global footprint and best-of-breed technologies to deliver comprehensive data availability and as-a-Service solutions by offering custom managed and monitored services with 24/7 live support. KeepItSafe’s secure enterprise-class data centers support virtual, physical, and cloud-to-cloud solutions with 256-bit encryption and multi-cloud scalability via a global network of service providers, system integrators, and cloud resellers. Find more at KeepItSafe.com and comment at @KeepItSafe.

KeepItSafe offers comprehensive cloud data availability solutions — contact us.

888 965 9988 | keepitsafe.com | sales@keepitsafe.com