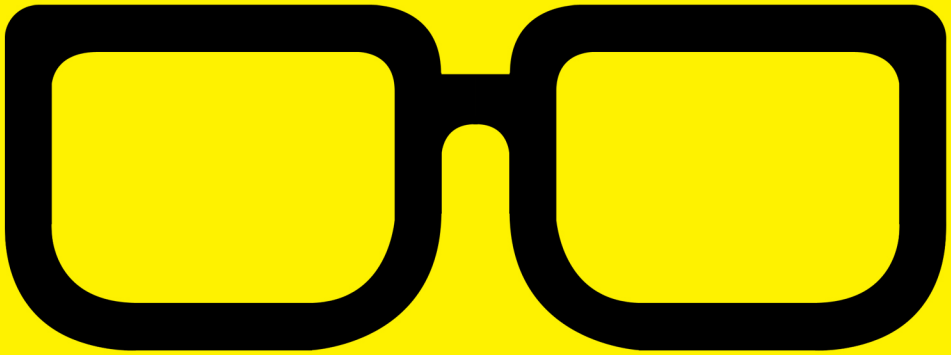
SPONSORED BY

SUSE®

GEEK GUIDE



SUSE Enterprise Storage 4

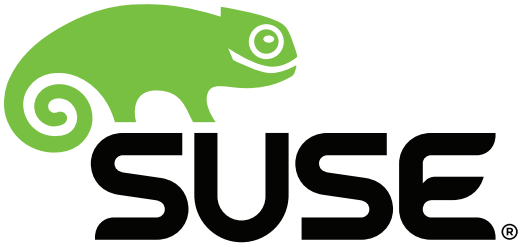
LINUX™ JOURNAL

Table of Contents

TED SCHMIDT is the Senior Project Manager and Product Owner of Digital Products for a consumer products development company. Ted has worked in Project and Product Management since before the agile movement began in 2001. He has managed project and product delivery for consumer goods, medical devices, electronics and telecommunication manufacturers for more than 20 years. When he is not immersed in product development, Ted writes novels and runs a small graphic design practice at http://floatingOrange.com. Ted has spoken at PMI conferences, and he blogs at http://floatingOrangeDesign.Tumblr.com.

GEEK GUIDES:
Mission-critical information for the most technical people on the planet.

About the Sponsor



SUSE, a pioneer in open-source software, provides reliable, interoperable Linux, cloud and storage infrastructure solutions that give enterprises greater control and flexibility. More than 20 years of engineering excellence, exceptional service and an unrivaled partner ecosystem power the products and support that help our customers manage complexity, reduce cost and confidently deliver mission-critical services. The lasting relationships we build allow us to adapt and deliver the smarter innovation they need to succeed—today and tomorrow.

SUSE Enterprise Storage 4

TED SCHMIDT

Introduction

I wrote a previous Geek Guide, titled *Ceph: Open-Source SDS*, that briefly introduced a Ceph-based, data storage management system called SUSE Enterprise Storage. Based on the response from readers of that ebook and given recent advancements in the maturity of SUSE Enterprise Storage (SES), it seemed logical to explore some of the features of SES more closely.

In this ebook, I review the characteristics of software-defined storage, along with its business benefits. Then, I explore the features of SUSE Enterprise Storage and how those capabilities can really help your organization leverage the benefits of open-source software-defined storage (SDS).

By separating storage management software capabilities from hardware, an enterprise can remove its dependency on proprietary software and any associated limitations.

A Quick Review of Software-Defined Storage

If you read the predecessor to this Geek Guide, you'll remember that software-defined storage is a technique for virtualizing storage capabilities on commodity hardware in an effort to reduce costs and improve efficiency. By separating storage management software capabilities from hardware, an enterprise can remove its dependency on proprietary software and any associated limitations. It also frees the organization to leverage the lower cost model of commodity hardware. This ends up being a key benefit of of any SDS solution: reducing capital expenditure (CAPEX).

More on the benefits later—first, let's do a quick review of the defining characteristics of an SDS solution. I've already mentioned that to be SDS (and this is a point on which both Gartner and IDC agree), a solution must be hardware-agnostic—that is, it must be able to use any brand of hardware. Otherwise, it just perpetuates the cost issue that comes from dependence on proprietary software.

In addition to hardware agnosticism, an SDS solution also has to provide the ability to establish policies for managing not only storage, but data services as well.

It must provide tagging of metadata for both storage and data services, as well as disaggregation of storage and data services. A true SDS solution also will provide automated management of storage and a self-service graphical UI as well. As you'll see later, SUSE Enterprise Storage provides a GUI that is informative and intuitive.

There are also a few optional features that SDS might provide, such as scaling out of storage, the ability to enhance specialized hardware functions and incremental build-outs of storage and data services. But those are not required for a solution to meet the criteria to be called "SDS".

One of the key business benefits of having an SDS solution comes from having a single management UI. Instead of having a different management center for each hardware vendor, and having to grapple with the different capabilities of each vendor's storage software, a single SDS solution helps simplify the management of an enterprise's storage and data services. You should expect to reduce operating expenses because you'll avoid having to train on new versions and releases, as well as reduce the risk that comes from your team having to interpret different data across different management consoles.

Another significant benefit is the reduction of CAPEX that comes from separating the purchase of hardware from any proprietary management software when the time comes to upgrade or add to your storage. With a single SDS solution in place, your enterprise is free to scale using commodity hardware without the associated capital outlay that typically comes with proprietary software. As I will show later, an SDS solution prevents newer hardware

from sitting latent and unused, as often happens with conventional block-storage solutions.

Because you are using commodity hardware, with SDS you can "borrow" capacity across arrays as demand fluctuates and avoid bottlenecks and single points of failure. Because Ceph provides shared storage capabilities across commodity storage devices, mirroring, replication and other needs won't create bottlenecks, and you don't need dedicated devices for storage arrays and storage area networks. In this way, SDS creates resiliency and high availability.

In summary, by implementing a software-defined storage solution, an enterprise is freed from the operational constraints and costs of having to use proprietary software to manage storage and data services. By leveraging the cost advantage of commodity hardware, managed through a single management UI, risk and CAPEX are reduced while a more scalable storage solution is provided. Now, let's take a look at how SUSE Enterprise Storage makes this all happen.

SUSE Enterprise Storage 4 Features and Capabilities

The first thing to understand about SUSE Enterprise Storage 4 (SES) is that it's built on Ceph. Ceph is the open-source, distributed object store and filesystem that was designed specifically to provide exceptional scalability, reliability and performance for object, block and filesystem storage in a single, integrated system. Not only is it open-source, Ceph is infinitely scalable and avoids single point of failure risk posed by proprietary systems that don't allow sharing.

Because it's built on Ceph, SUSE offers a much more

scalable solution that performs much better than legacy block-storage solutions. With SUSE, data size is variable, as opposed to a traditional block-storage solution where data is divided into fixed-length blocks. And also with SUSE, that data is distributed across the system and its location is tracked using the CRUSH (Controlled Replication Under Scalable Hashing) algorithm. In a conventional block-storage system, fixed-length blocks of data are tracked by a central management system. This leads to bottlenecks as the centralized system breaks up data into evenly sized blocks and then looks for available slots into which it can insert the data.

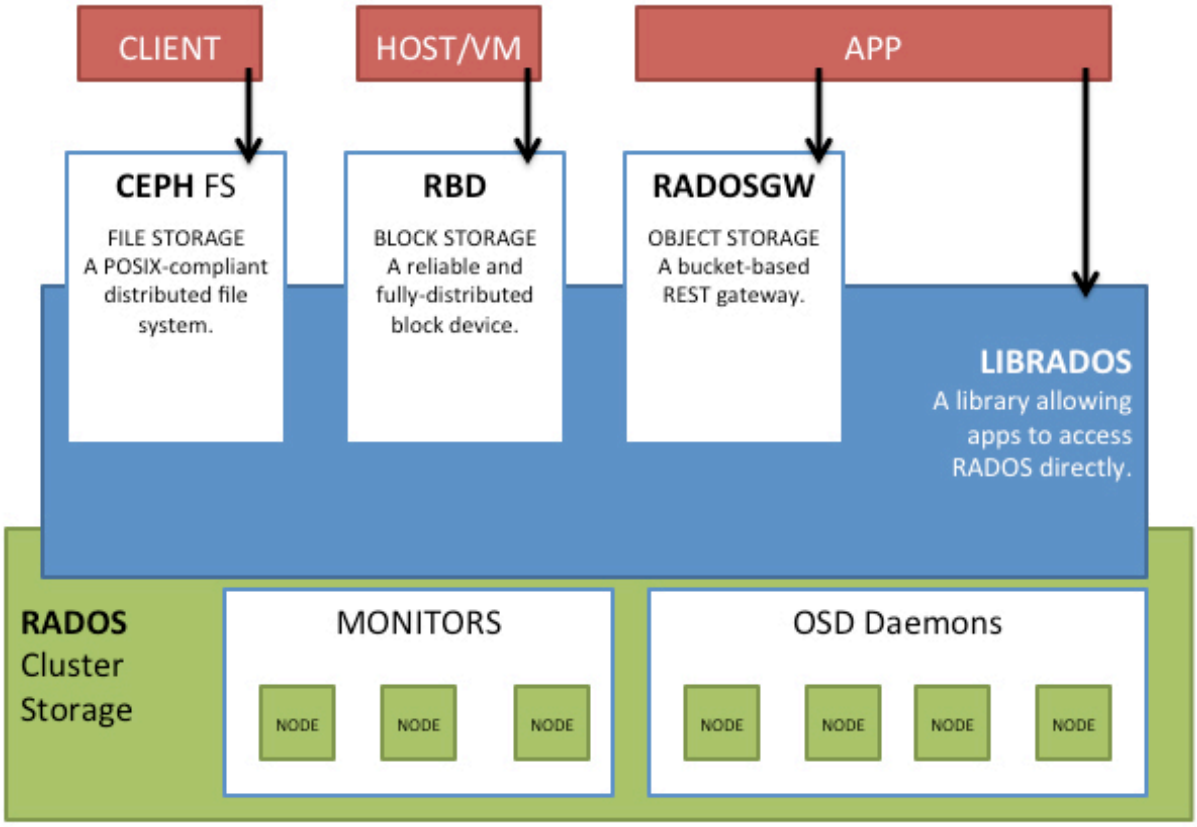


FIGURE 1. SUSE Enterprise Storage Architecture

The RADOS-based storage cluster in SES is composed of a monitor to maintain maps of the cluster state and ensure high-availability, at least two Object Storage Devices (OSDs) to store data and a Metadata Server (MDS) to store metadata for any filesystem clients. All the dæmons and clients in SES have knowledge of the cluster topology, and the OSDs are cluster-aware, which allows them to access CPU and RAM of cluster nodes without causing bottlenecks. And, because clients can access OSDs directly, performance and system capacity are both improved.

Whenever a file is opened in a traditional block-storage solution, the centralized management system searches for all the related blocks, gathers them and reassembles them into a single file. Even if this works for small, localized systems, what happens when new storage is added? Since stored data in a traditional block-storage system is hardly ever moved from its storage location, adding any new storage creates terrible imbalances in the system, as most of the new storage simply remains empty and un-utilized. This is hugely inefficient and slow, especially for big data and big networks, compared to SUSE's Ceph-based object approach where the CRUSH algorithm takes care of managing the location of distributed data.

CRUSH uses the established data placement policy and a map of the storage cluster devices to determine how to store and retrieve data. Because CRUSH is distributed across the system, any node can find any data object, which is much more efficient than relying on a centralized brain. Not only does CRUSH provide this advantage, but it relies on very little metadata.

Any client that wants to retrieve data from the array simply sends a request to a placement group. In response, the primary OSD serves that data back to the placement group for the client. Think of it as a waiter placing an order sheet on the pass-through window to the kitchen in a restaurant. The cook picks up the order sheet, prepares the food and places the prepared food back on the pass-through window for the waiter to retrieve. The waiter doesn't have to know where any of the ingredients actually are stored in the kitchen, just like the client has no knowledge of the specific storage location of the data—the location of the data is stored within the network itself.

The OSDs not only process client requests, they also check on each other and send monitoring data back to the monitor node, which then updates the CRUSH map to reflect that status. This distributed approach then removes the opportunity for bottlenecks that are present in centralized block-storage systems.

As mentioned earlier, by using commodity servers and drives and keeping them separate from the management system, SUSE Enterprise Storage 4 allows an enterprise to build a highly scalable storage system that avoids bottlenecks and single points of failure, and provides a consolidated, intuitive GUI for management that is not proprietary. So, when your enterprise needs to extend its hardware, you don't need to worry about incurring additional licensing or training costs.

Ask senior IT decision-makers and most of them will agree that scaling to meet data growth needs and fluctuations, cost and flexibility are the key issues facing them when it

And because it provides data access at the filesystem, object and block levels, SES is extremely flexible, allowing you to connect block devices to virtual machines.

comes to data storage. Offering a solution that is cost-effective while providing the ability to scale with business needs quickly and easily answers these concerns, and that is exactly what SES offers in its features.

As I've discussed, being built on Ceph, SES has several features that provide real business benefits, the first of which is high availability. Remember that Ceph eliminates any single point of failure, because Ceph distributes the workload across multiple devices. SES doesn't rely on dedicated devices, but rather shared capacity that is constantly managed.

Another feature offered by SES is scalability. Using commodity hardware, the CRUSH algorithm helps SDS balance storage across the entire storage cluster. And because it provides data access at the filesystem, object and block levels, SES is extremely flexible, allowing you to connect block devices to virtual machines.

Getting Started with SES

According to SUSE's documentation, minimum and recommended configurations are fairly simple. At a minimum, SUSE recommends four OSD storage nodes

and a separate management node. Each of the OSDs should have 1GB of RAM for every TB of capacity and be able to run at 1.5GHz for each OSD storage node on each OSD. Each OSD storage node should possess a dedicated OS disk. The OSD journal can reside on an OSD disk, and there can be 32 OSDs on every storage cluster. It's worth noting that monitor, gateway and MDS nodes can all reside on OSD nodes. The separate management node should be built with 4GB of RAM, four cores and a terabyte of storage capacity.

That's the minimum configuration just to get started. Once you get serious, you'll need to bump things up a bit for a production environment with seven OSDs, balanced at around 15% each. Each OSD will want 1.5GB for every terabyte of capacity at speeds of 2GHz, versus 1.5GHz. You'll also want three dedicated monitors that have 4GB of RAM, four cores and RAID 1 SSDs for disk. A separate management node will be similarly equipped. SUSE recommends using SSDs for both infrastructure nodes and for journaling (at a 6 to 1 ratio for SSD journals to every OSD), and RAID 1 disks for OSD storage, monitors and the management node. Clearly, minimums for a production environment are a bit more than the basics, so research and read SUSE's recommendations before jumping in.

You can find more on SUSE configuration here: https://www.suse.com/docrep/documents/y13l1qoto9/SUSE_enterprise_storage_configurations.pdf.

Once you've built and configured your SES Ceph storage cluster, it pretty much runs and manages

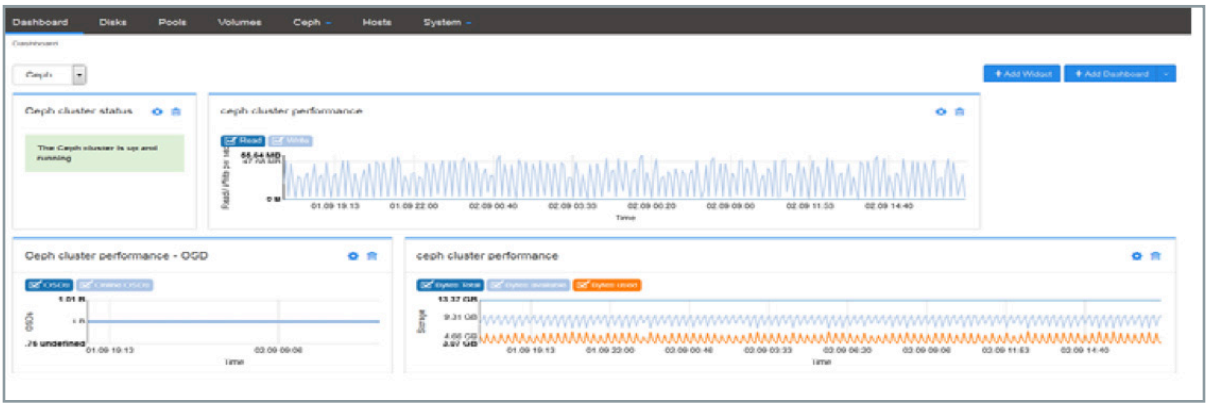


FIGURE 2. OpenATTIC Graphical UI

itself. Most of your time will be spent up front in building the hardware assembling the OSDs, monitors and management node, and configuring the CRUSH algorithm. CRUSH can be customized to fit your network's needs and demands, and a little tweaking at the beginning can result in even greater benefits.

Tying all these features together is the open-source management system of SES, which is the result of a recent collaboration between SUSE and OpenATTIC. SES actually requires very little intervention by constantly monitoring data utilization and calculating the best possible data placement using intelligent algorithms. This creates the best performance possible while minimizing the need for manual intervention. That being said, one of the biggest advantages that SUSE Enterprise Storage 4 provides is the OpenATTIC GUI. Not only does OpenATTIC provide features such as multi-server management and mirroring, but the latest version, built on AngularJS and Bootstrap frameworks, now makes use of REST APIs,

instead of XML-RPC.

So, while the open-source management system for SES provides effective background monitoring, it also reports on the performance of your data storage. Using intuitive graphing, the OpenATTIC GUI provides quick access to overall status and utilization, condition of your OSDs and how your input/output operations per second (IOPS) trends over time. Complete access to information on your disks, pools and volumes are only a click away. You also can drill down and look at command logs from a variety of sorted views; OpenATTIC is rich in summary and detailed management data.

Built on Ceph, SUSE Enterprise Storage 4 provides a single, unified GUI, powered by OpenATTIC, to manage your storage. This in itself helps reduce OPEX. But, there are other clear business benefits. SES is highly scalable, providing a definite business advantage as your data storage needs shift and grow. Because you aren't committed to any single vendor for your storage needs and can make use of commodity hardware, your data storage solution is flexible and low-cost.

Conclusion

I've reviewed software-defined storage and its benefits, and I've taken a closer look at how the features of SUSE Enterprise Storage, built on Ceph, deliver those benefits through a distributed, object-based architecture. By spreading the management load across object storage devices, SES avoids single point of failure and bottlenecks, and the common management GUI reduces OPEX costs.

SUSE Enterprise Storage (SES) delivers true open-source software-defined storage that is built on Ceph and the CRUSH algorithm. It offers an excellent GUI that provides performance insights to help your enterprise understand and manage your object-based storage solution. Configured according to SUSE guidelines, SES can handle your big data and big network needs for scalability, high availability and flexibility, all in an efficient, self-managing solution.■