UNPRECEDENTED PERFORMANCE, SCALE, AND SAVINGS

3 Powerful Benefits of Modernizing Your Splunk Environment

An MSP reference design leveraging the power of virtualization, Pure Storage®, and world-class Splunk engineering expertise

MSP VARIANT REFERENCE DESIGN

In September 2020, Kinney Group, Inc. (KGI), in collaboration with Pure Storage, released its Splunk Reference Design for FlashStack. This design makes use of Pure Storage FlashArray and FlashBlade technologies, combined with Splunk SmartStore, to create a compelling on-premise solution for operating the Splunk platform at scale.

The Pure Storage FlashStack offering is a Cisco Validated Design (CVD) infrastructure solution that combines FlashArray and/or FlashBlade storage platforms with Cisco UCS compute technology and Cisco Nexus switching technology. The FlashStack CVD has been thoroughly tested and validated for performance in interoperability. Kinney Group's reference design for FlashStack leverages the architecture testing and validation that is provided by Cisco via their CVD certification.

The Splunk Reference Design for FlashStack released by KGI helps customers achieve three key outcomes. These include:

1. **Unmatched performance.** The reference design makes full use of the compute, storage, and network performance enabled by the FlashStack CVD to produce an infrastructure solution that optimizes the performance of the Splunk platform.
2. **Simplified scaling.** KGI's overall design combines the power of the FlashStack CVD with the agility enabled by VMware vSphere and Splunk SmartStore software technologies. The end result is a design that enables customers to scale their Splunk implementations quickly and easily.
3. **Lower total cost of ownership.** The combination of FlashStack, virtualization techniques enabled by vSphere, and hybrid data storage techniques enabled by SmartStore has produced a Splunk architecture solution that minimizes TCO. Additionally, KGI has made extensive use of automation to further drive down the costs associated with deploying and operating Splunk at scale.

A Managed Services Provider Approach

The pages that follow in this white paper describe an approach that applies KGI's reference design techniques used with FlashStack in a manner that aligns strategies that many managed services providers (MSPs) and hosting providers apply to the infrastructure architectures. KGI has labeled this reference design as the "MSP variant" of the previously described design that is based on the FlashStack CVD.

MSPs and hosting providers (and companies that operate like them) consistently take architecture and operational approaches that are purpose built to reduce costs associated with acquisition, operation, and architecture sustainment. At a high-level, MSPs pursue the following strategies:

REDUCE CAPITAL EXPENDITURE COSTS

MSPs and hosting providers consistently will pursue infrastructure technologies and architectures that minimize capital expenditures (capex). To reduce capex costs specific to infrastructure hardware, MSPs make effective use of compute and network platforms that provide performance, scalability, and reliability without the cost of a brand name.

MAXIMIZE NETWORK DESIGN FLEXIBILITY

Infrastructure and solution capabilities that are provided by MSPs, hosting providers, and cloud providers are all accessed remotely by end-users over wide area network (WAN) connections. To optimize WAN performance and the end-user experience, MSPs and hosting providers consistently pursue network designs and technologies that maximize flexibility while not sacrificing performance.

MINIMIZE OPERATIONAL COSTS

MSPs, hosting providers, and cloud operators are very attentive to minimizing operations and sustainment costs in ways that do not compromise performance and reliability. These operators are effective at minimizing all operations costs – from cost elements as mundane as electricity and cooling to more sophisticated considerations such as processor core density and switch port densities, these providers ruthlessly pursue driving out operational costs associated with their architectures.

Additionally, these providers pursue all cost reductions associated with personnel and with ongoing infrastructure technology sustainment. For example, many providers will maintain their infrastructures with internal resources combined with “mail-in, carry-in” approaches to hardware warranties. These cost-reduction approaches are in contrast to traditional maintenance and warranty offerings that typically carry annual costs that are 8-12% of the original capex investment in hardware.

KGI’s Approach to an MSP Variant Design

The vision of an MSP Variant design is to deliver the performance, scalability, and TCO outcomes of the Splunk Reference Design for FlashStack in a manner that aligns with the operational and cost-reduction objectives of MSPs, hosting providers, and cloud operators. To achieve this vision in the MSP Variant design, KGI pursued the following:

Continue to leverage the power of virtualization and automation that was achieved in the originated FlashStack-based reference design. To achieve this objective in its MSP Variant design, KGI applied all of the software approaches that were successfully used in the Splunk Reference Design for FlashStack.

Pursuit of a design that minimizes acquisition costs and operational costs at the compute and storage layers, while not sacrificing performance, scalability, and resiliency. To achieve this objective, KGI pursued three key design attributes that measurably reduced costs associated with compute and storage. These included the following:

Make use of Intel-branded servers in lieu of premium priced Cisco UCS servers. This approach enables the acquisition of required compute power at a lower cost. As well, KGI saw reduced operational costs associated with Intel servers when compared to comparably configured Cisco UCS servers.

As was done in the original FlashStack design, continue to leverage the power of Splunk SmartStore as a means for driving out storage costs. SmartStore enables MSPs and hosting operators to make use of inexpensive cloud-based storage without sacrificing performance.

Removal of the FlashArray platform from the FlashStack design by using clustered, server-based SSDs to host the SmartStore cache. This approach drove down total storage costs without reducing any of the gains achieved in the original FlashStack design.

Achieve maximum network design flexibility and performance via a switching platform that allows full access to the switches operating system. To achieve this objective in the MSP Variant design, KGI replaced the Cisco Nexus platform with technology from Arista Networks that is widely favored by hosting and cloud providers for its flexibility and lower costs of acquisition and operation.

Conclusions

The pages that follow will provide deeper insights into the performance and cost results achieved in the MSP Variant reference design. Similar to the great results achieved in KGI's original Splunk Reference Design for FlashStack, the MSP Variant design produces equal (or sometimes better) performance and operational results, while at the same time achieving the cost reduction and architecture flexibilities that MSP and hosting providers seek.

With its reference designs for Splunk, KGI always seeks to maximize performance and scale, enable simplicity, and reduce total cost of ownership. KGI believes that it can achieve these objectives in ways that best align with individual customer organization's strategic objectives.

For customers that have significant investments in Cisco compute and network technologies, the original Splunk Reference Design for FlashStack will likely be the best fit. For MSPs, hosting providers, cloud operators, and end-user organizations that approach their architectures like these players, the MSP Variant design provides a compelling approach for achieving optimal performance and cost results associated with operating Splunk technologies at scale.

Key Benefit #1:
Unmatched Performance

The beauty of this reference design lies in the unmatched performance provided by combining VMWare Virtualization, Splunk SmartStore, and Kinney Group's advanced Splunk configuration tuning in a virtualized environment.

PureStorage FlashBlade supports file and object storage, producing a seamless integration with Splunk SmartStore. These technologies provide an all-flash performance, even for data that would have been traditionally rolled into cold buckets on slower storage tiers. Kinney Group optimizations enable rapid ingest and quick searches even at high volume, and testing showed the reference design can easily ingest up to 4x the Splunk-recommended limit. That means a Splunk-recommended architecture for 500 GB of daily ingest can handle 2 TB or more. In fact, testing showed that a sustainable result of 8x the recommended limit (4 TB/day) is possible.

Optimizing Splunk for Lightning Fast Search

Kinney Group's engineering expertise in optimizing Splunk enables users to ingest more data, more quickly. Optimization and fine tuning of the environment yields astonishing results. Splunk searches on traditional, distributed scale-out architectures lead to significant performance degradation as data ages. As it ages, data is tiered to cheaper and lower-performance storage tiers in cold buckets, significantly impacting search performance. This storage approach is especially impractical when responding to search requests related to regulatory or compliance requirements, cybersecurity, and legal discovery—all of which demand information beyond the most immediate data. Utilizing SmartStore with FlashBlade, however, provides all-flash performance with high bandwidth and parallelism for data operations and searches outside of the SmartStore cache. It also ensures that you can efficiently complete critical, non-repetitive tasks while supporting the bursting of SmartStore indexers. By Splunk best practices, high search execution latency should be avoided and can cause a cascading degradation in performance. At the highest levels of data throughput tested in the validation of this design, disk latency never exceeded 2ms, and Input/Output Operations Per Second (IOPS) remained flat.

4x

data ingest versus Splunk's guidance

For data ingest of 500GB/day, Splunk recommends a minimum of 5 indexers* (100GB per indexer), meaning you would need 20 indexers or more for 2TB of volume. This reference design allows for 2TB+ of daily ingest using only 5 indexers — a 75% decrease in hardware requirements.

*https://docs.splunk.com/Documentation/ES/5.2.0/Install/DeploymentPlanning

FlashBlade + SmartStore

Faster insights from searches, regardless of data age

Increased availability of data thanks to N+2 Erasure Coding protections

Simplified management of Splunk clusters (at any scale) without the need for data migration

99%+ faster node addition and data rebalancing

Faster data access at scale

Faster insights from searches, regardless of data age

No indexer data replication required

Compression and capacity combine to allow you to keep data searchable longer on lightning fast flash storage

Optimizing Splunk for Lightning Fast Security Workloads

Using Splunk Enterprise Security (ES) “off the shelf,” there are a number of inefficiencies in search configuration. In the testing and validation of this Reference Design, Kinney Group was able to tune ES to avoid skipped searches while maintaining the level of searches in the environment. Splunk will often skip scheduled searches — as a result of high latency that Splunk is not able to overcome — by postponing or rescheduling the search or searches. This was accomplished, in part, by including updated timing of searches and increasing search slots in the software. (See the “Enterprise Security Tuning” section of this document for details.)

The net result is an environment with such precise software tuning and hardware engineering that you’ll imagine the sound of a perfect Formula-1 racing engine every time you walk by your server room.

Enabling Data Security without Hindering Performance

In a traditional Splunk environment, enabling data security introduces various considerations that significantly impact performance. Pure Storage FlashBlade supports native data encryption while still maintaining incredible single chassis performance of 1.5 million IOPS and 15 gigabytes per second (GB/s) of throughput at consistently low latency.

We hate to say “faster, better, cheaper,” but…

We know how tired the “faster, better, cheaper” trope is, but the reality simply can’t be avoided. This unmatched performance doesn’t come with the soul-crushing price tag you’d expect. Rather, we’ve engineered a solution that allows you to reduce footprint and impact the total cost of ownership (TCO) in a way that demands further inspection — you’ll save on capital expenses, operating expenses, and who knows how much on aspirin.

Key Benefit #2:

Simplified Scaling

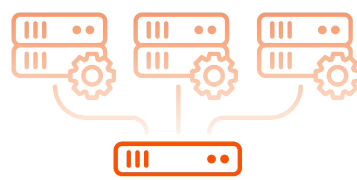
Accommodating scale is an ever-present struggle for IT teams and data center operators — providing sufficient infrastructure to facilitate more demanding requirements such as increasing compute, storage, and network needs. Complexities introduced by Splunk's specialized data ingest requirements only make the situation more challenging (not to mention costly).

The true benefit of scaling is realized not just when future growth is enabled, but when front-end requirements can be met with less hardware, expense, and footprint. Scaling only matters if you can grow from a reasonable starting point. The Kinney Group PureStorage Reference Design empowers uses to achieve better performance at scale from their Splunk environment while requiring 75% less hardware.

Managing growth requires systems and strategies that cost-effectively and efficiently support scale. While traditional data center models rely on prohibitive infrastructure requirements in order to scale (square footage requirements, ballooning engineering and operational costs, and a never-ending list of hardware requirements and purchases), FlashBlade allows incredible scaling in a smaller form factor. Cloud infrastructure provides great scaling, but growing out an existing Splunk cloud architecture is costly, complex, and operationally challenging. The Kinney Group PureStorage Reference Design is a powerful and elegant solution that enables data centers and Splunk solutions to "grow in place."

The Power of Virtualized Scaling

Splunk excels at extracting hidden value from ever-growing Machine Data. This workload, however, requires massive storage capacity, so infrastructure needs to be flexible and scalable, while also providing a linear performance increase alongside that scaling. Simply put, more data means more storage and computing power needs.



Scaling Splunk by leveraging VMware virtualization

Scaling compute & storage is usually solved by physical hardware additions. Virtualization allows you to utilize more of your hardware's capabilities by creating multiple virtual machines on a single piece of physical hardware.

Virtualization reduces footprint, TCO, and time to perform critical tasks.

While the traditional approach of using physical servers for deployment is certainly an option, utilizing virtual machines on scalable hardware solutions allows you to save time, space, and budget while being able to scale and grow “on the fly” as required.

Typical Splunk deployments utilize a handful of components at their core — Forwarders, Indexers, and Search Heads. Forwarders collect and forward data (lightweight, not very resource intensive). Indexers store and retrieve data from storage, making them CPU and disk I/O dependent. Search Heads then search for information across the various indexers, and are usually CPU and memory intensive.

By properly utilizing virtual machines, the Kinney Group PureStorage Reference Design allows users to scale resources to match the increasing demands of these components.

Physical Scaling that Doesn’t Grind Operations to a Halt

Modern data centers are looking less and less like giant warehouses of server racks and becoming more distributed, but the basics of traditional data center growth have experienced little disruption, depending heavily on increasing the number of servers, racks, electrical distribution, and space required to accommodate growth.

Utilizing PureStorage FlashBlade enables cloud-like simplicity and agility with consistent high performance and control. The primary way FlashBlade enables grow in place scale is by allowing massive physical expansion in a single chassis by adding “blades,” each of which increases capacity and performance without requiring an ever-growing footprint. Rather than shutting down data center operations to scale out by adding new servers and bringing them online alongside existing infrastructure, the FlashBlade solution allows users to grow in place. PureStorage FlashBlade provides up to 792 Terabytes of raw storage in a single 4 rack unit (RU) chassis. Storage is further optimized by using SmartStore, which removes the need for indexer replication (typically a factor of 2 for all data). The total system can grow to ten chassis. FlashBlade also supports in-service hardware and software updates, so scaling up and scaling out won’t interrupt operations.

Meet Any Compliance Requirement with Unlimited Scaling

Splunk SmartStore makes the daunting task of data retention simple for organizations that have compliance or organizational obligations to retain data. This PureStorage architecture supports up to 10 FlashBlade chassis, potentially representing years of data even for high-ingest systems.

How Does FlashBlade Enable Scale?

FlashBlade is built to scale every dimension of the system effortlessly and linearly—IOPS performance, bandwidth, metadata performance, NVRAM and client connections.

The blade is the scaling unit for FlashBlade. Each blade marries raw NAND flash with system-on-a-chip processing.

FlashBlade has been designed so that anyone can install it. Scale-out is simple, instant and online; to add capacity, you simply add blades — up to 15 per 4U chassis.

Key Benefit #3:

Lower Total Cost of Ownership

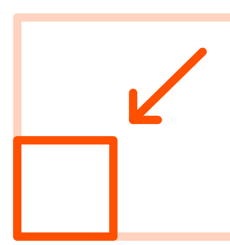
The distributed data center model provides high availability by replicating data, but effectively eliminates any benefits gained from Splunk data compression by increasing storage requirements. Co-locating storage and compute means when you need more storage, you have to add both compute and storage. To further increase total cost of ownership (TCO), Splunk indexers with a distributed scale-out architecture usually have more servers with less storage to minimize the amount of data and time associated with server maintenance and failures.

In short, the old-school, conservative approach of an ever-growing physical data center comes with incredible expense and tremendous financial risk.

Reduce Server Counts

Splunk recommendations for an Enterprise Security (ES) deployment with a 2 TB daily ingest call for up to 20 indexers. Based on validated testing with this Reference Design, we were able to achieve similar or better performance with only 5 indexers. This 4x improvement over Splunk recommendations represents an incredible cost savings for organizations in year one alone.

Using SmartStore with FlashBlade, Kinney Group's Reference Design lowers the storage and compute requirements when compared to Splunk's classic, "bare metal" storage architecture. With this approach, indexers can be sized based on ingest rates and concurrent search volumes instead of worrying about storage. Additionally, SmartStore only requires storage of a single copy of warm data, and FlashBlade further reduces storage requirements for the object tier by 30–40% through data compression.



Reduced Footprint = Reduced Costs

This validated architecture proves that you can run Splunk more efficiently and with better results, utilizing only a fraction of Splunk's recommendations

Increased performance with just 5 indexers vs. the recommended 20

Impact TCO through storage performance, availability, scalability… all while providing unparalleled results and reducing risk

Reduce Storage Costs by 62%

The impact is even greater when you consider the topic of storage efficiency using the Kinney Group PureStorage Reference Design on FlashBlade. Storage efficiency — fitting more data into less raw flash — is a key contributor to reducing the amount of physical storage that you must purchase, deploy, power, and cool.

In ESG's Economic Validation report, "Validating the Economics of Improved Storage Efficiency with Pure Storage," the results show that Pure saved financial services organizations up to 59% in TCO, and healthcare and government organizations up to 62% through storage efficiencies alone.

Reducing CapEx AND OpEx: Considering Total Financial Impact

While a reduction in the capital costs associated with server and storage acquisition are compelling, those costs typically contribute only 20% (or less) to a 3-year server TCO, with management and other OpEx contributing the remaining 80%.

How does this reference design decrease operating expenses? The short answer is that a smaller footprint means a reduction across the board in the month-to-month and year-over-year expenses hidden in operating a data center — costs like power consumption and other utilities, preventative and predictive maintenance, connectivity, and staffing, to name a few.

With this reference design, you'll impact bottom-line savings through storage performance, availability, scalability, and performance — providing the potential to grow revenue streams and lower costs. You'll significantly reduce overhead by reducing the number of servers required to drive your Splunk ES environment, while simultaneously providing unparalleled results and reducing security risk. And, especially of importance, you'll substantially reduce operating expenses associated with a sprawling data center footprint.

Total Cost of Ownership (TCO) is a complex subject, to be sure. The bottom line is that implementing a powerful, scalable compute and storage solution such as FlashBlade technology in conjunction with SmartStore in a Kinney Group-tuned Splunk environment provides both immediate and long-term financial benefits for your organization.

Technical Solution Overview

Kinney Group is a cloud solutions integrator that designs, builds, and integrates IT infrastructure solutions for some of the most demanding government agencies and commercial organizations. By leveraging next-generation technologies, and adopting proven engineering practices and agile development principles, we create custom solutions and world-class environments for data.

Kinney Group, Inc. (KGI) is a leading provider of Splunk platform solutions. Our team has experience working with deployments of all sizes, various stages of execution, and across a variety of use cases. We've helped companies identify end goals, develop and implement Splunk, grow and optimize their environment, and everything in between.

KGI is leading the way in designing a virtualized reference architecture that can be utilized by Pure Storage customers as a guideline for building their own resilient Splunk Enterprise Security (ES) environment. For this endeavor, KGI has leaned on their experience with Pure Storage, Intel, and Arista networks which incorporates Intel based servers utilizing Intel's Optane Storage, and Pure's FlashBlade.

This paper is intended to provide a framework for designing and sizing a high-performance, scalable, and resilient Splunk platform for Splunk Enterprise Security. VMware is a leading platform for server virtualization. Pure Storage is a leading all-flash array provider that supports Splunk Smart Store. Using a combination of VMware, Pure Storage, and Splunk SmartStore, customers can reduce storage complexity and datacenter footprint while maintaining platform performance, resiliency, and efficiency.

Splunk hardware specifications recommend 20 indexers for 2TB daily ingest. We observed acceptable user experience with only 5 indexers running all Pure storage. That's a 75% improvement over Splunk's recommendation.

The below table shows our key findings:

Daily Ingest	Number of Indexers	Disk Latency	Search Latency	Skipped Searches
2TB	5	1.5 ms	<2 seconds	0

GOALS AND OBJECTIVES

The goal of this reference architecture is to showcase the scalability, performance, manageability, and simplicity of the virtualized platform solution for a large scale Splunk Enterprise Security deployment.

The key objectives for this reference architecture:

- Design repeatable architecture that can be implemented quickly in production sites
- Utilize VMware to reduce datacenter footprint and scale environment quickly
- Utilize SmartStore to take advantage of shared object storage to improve operational efficiency and reduce overall disk requirements for the environment

AUDIENCE

The target audience for this document includes, but is not limited to, system administrators, storage administrators, IT managers, system architects, sales engineers, field consultants, professional services, and partners who are looking to design and deploy Splunk Enterprise in a virtualized environment. A working knowledge of Splunk, VMware, Linux, server, storage, and networking is assumed, but is not a prerequisite to read this document.

REFERENCE ARCHITECTURE DESIGN PRINCIPLES

The guiding principles for implementing this reference architecture:

Simple Using pre-built images and apps, we minimize the amount of manual configuration required.

Secure Combination of solid security architecture and compliance concepts built in the design.

Available By using a combination of indexer clustering, Splunk Smart Store, and Pure Storage, we can create an environment that needs nearly zero downtime for upgrades and updates and is fault tolerant to unexpected failures.

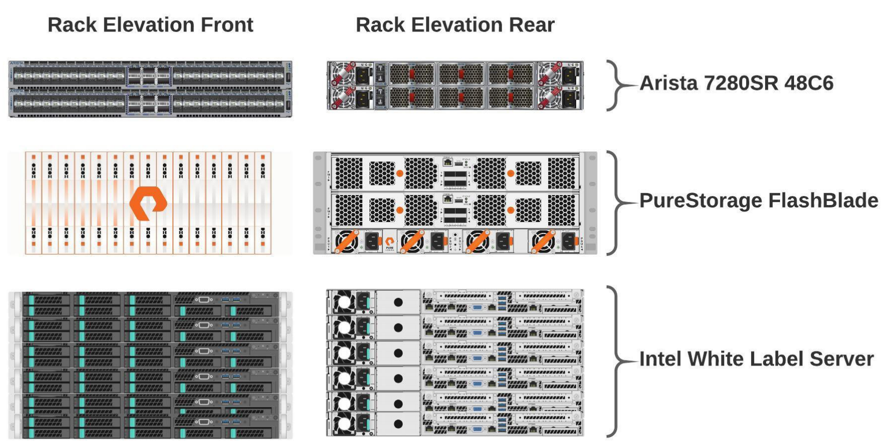
Efficient By utilizing VMware, SmartStore, and Pure Storage, we reduce the overall required datacenter footprint, saving power and cooling costs.

Cost Effective Combination of technologies drives up flexibility and drives down costs.

Elasticity Reference design can be scaled based on customer's daily data volume.

Solution Design

DESIGN TOPOLOGY



VIRTUAL SERVER CONFIGURATION

For Enterprise Security (ES), Splunk recommends sizing based on 80 to 100 GB ingest per indexer, per day. This means an ES deployment with 2 TB daily ingest will require up to 20 indexers. Based on our test using this reference design and tuning, we were able to achieve similar, if not better, performance using 75% less hardware. This reference design will require no more than 5 virtualized indexers to support an ES deployment with 2 TB daily ingest with up to 24 active users.

Component	Description	Count
Indexer	16 vCPU 64 GB vRAM 250 GB Local Storage	5
Search Head (Enterprise Security)	16 vCPU 64 GB vRAM 200 GB Local Storage	1 ES 1 non-ES
Cluster Master	12 vCPU 32 GB vRAM 200 GB Local Storage	1
Deployer/Deployment Server	12 vCPU 12 GB vRAM 200 GB Local Storage	1

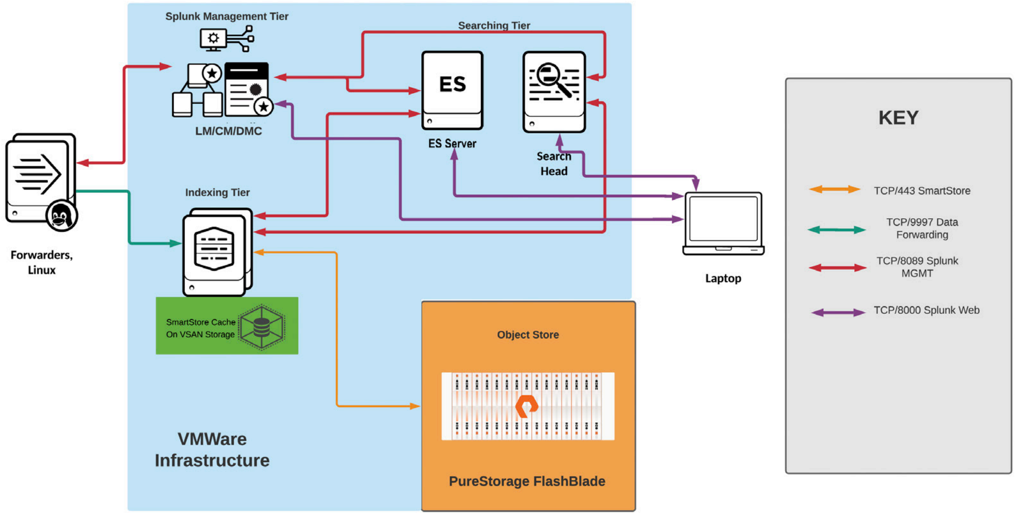
All servers will be run on a VMware stack hosted on the Intel white label servers.

PHYSICAL TOPOLOGY

Transcending the conventional model of bare metal installs, this solution for Splunk involves all virtual machines. The primary reason for choosing virtual machines is to allow for flexible workload positioning and scale out. By leveraging virtualization it is possible to rapidly scale the compute layer, either at a resources per machine level or number of machines to match your required workloads.

This reference design allows you to leverage an industry-proven, fully-documented hardware configuration to support your Splunk environment. By using Pure as the shared storage backbone, you receive the benefits of highly performant storage along with the business benefits of evergreen storage and non-disruptive upgrades.

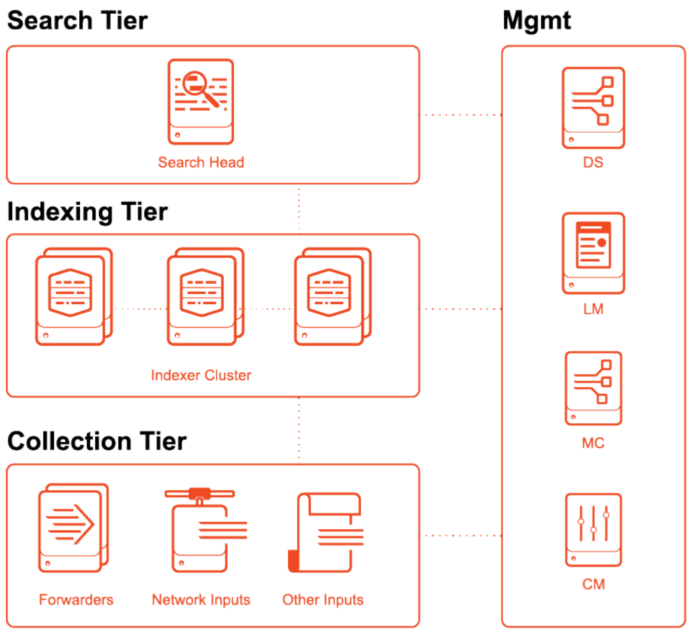
This solution is comprised of six Intel-powered white label servers containing Intel's Optane drives and software (VMWare Vsphere, Splunk Core & ES, Pure Storage GUI, Purity, CentOS Linux). The following diagram shows architecture of how the computer and storage of solution work with each other:



SPLUNK ARCHITECTURE

The architecture chosen for this solution comes from Splunk's Validated Architectures. For more information about the architecture chosen, please see https://www.splunk.com/pdfs/technical-briefs/splunk-validated-architectures.pdf

The design makes use of a single search head to run Splunk Enterprise and a separate search head for Splunk Enterprise Security, and multiple indexers in a cluster.



VMWARE BEST PRACTICES

The following configuration guides were used as a starting point:

- Performance Best Practices for VMware vSphere® 6.7 (https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/techpaper/performance/vsphere-esxi-vcenter-server-67-performance-best-practices.pdf)
- Deploying Splunk Enterprise Inside Virtual Environments (https://www.splunk.com/pdfs/technical-briefs/splunk-deploying-vmware-tech-brief.pdf)

Based on our knowledge and experience with the Splunk platform, some of these configurations were further refined to achieve optimal data ingest and search performance.

PERFORMANCE REPORTING APP

The Search Head includes a Kinney Group-built app to monitor the overall health of the Pure Storage array, along with measuring and reporting on performance metrics from the Splunk servers. The app will be used in conjunction with pre-built searches from the Splunk Monitoring Console.

Test Configuration

TEST OVERVIEW

We are able to confirm the validity of a Splunk architecture utilizing virtualized Splunk indexers, Pure Storage, and Splunk SmartStore. This reference architecture is capable of handling a standard Splunk Enterprise Security load with a daily ingest of up to 2TB.

We were able to test this theory by loading a set amount of machine data into the indexing layer. The replication factor and search factor for the index cluster were both set to 3, and Splunk SmartStore was enabled. To generate search load, data model acceleration was enabled on all data models with correlation searches scheduled to run through the duration of the test.

This reference design includes Ansible playbooks to automate deploying the VMware virtual servers, CentOS 7 servers, and Splunk Enterprise software.

HARDWARE USED

As shown in the physical topology, the test lab used Intel Optane as VMware VSAN for Splunk hot buckets and smartstore cache, FlashBlade for Splunk SmartStore, and Intel white label servers as compute.

SOFTWARE USED:

The following software packages are used in this design:

- Splunk 8.0.3
- Splunk Enterprise Security (latest version)
- Splunk_TA_cisco-asa (latest version)
- Splunk_TA_cisco-esa (latest version)
- Splunk_TA_cisco-wsa (latest version)
- Splunk_TA_isc-bind (latest version)
- Splunk_TA_mcafee (latest version)
- TA-crowdstrike (latest version)
- TA-ps_flashblade (latest version)
- splunk_app_gogen (version 0.5)

VIRTUALIZATION SETTINGS:

Splunk has provided recommendations for virtualization in deploying Splunk Enterprise inside virtual environments. All these recommendations along with performance best practices guide for vSphere were followed while provisioning VM and allocating storage. All Splunk VMs ran CentOS 7.

The following guide was followed for all VMware configurations: https://storagehub.vmware.com/t/vmware-vsan/splunk-on-vmware-vsan/splunk-virtual-machine-configuration/

INDEX VOLUME AND SMARTSTORE CONFIGURATION

A single storage volume was configured on the indexers This volume was set to 250 GB. We kept the cache size low to force eviction and cause searches to go back to SmartStore more frequently. To eliminate cache thrash, SmartStore cache can be adjusted to be a higher number. Splunk recommends this cache size to be equal to or greater than 90 days for Enterprise Security deployment.

TESTING PROCEDURE:

Testing was designed to mimic a real-world customer environment with a single indexer cluster and Enterprise Security running on a single search head.

1. Splunk Environment consisting of 1 search head, 5 indexers, and 1 cluster master (see Design Topology for virtualized server sizing)
2. Configure Splunk SmartStore to utilize Pure Storage
3. Configure Gogen on universal forwarders to generate a level of data from the below chart
4. Enable Splunk ES data model acceleration on all data models.
5. Enable up to 11 correlation searches that run on at least a 1-hour period
6. Utilize KGI-built monitoring app, PureStorage-TA, and monitoring console to measure health of data ingestion queues, search scheduler skip ratio, search latency, storage IOPs, and disk latency.

values(sourcetype)
WinEventLog
access_combined
cisco:esa:amp
cisco:esa:authentication
cisco:esa:http
crowdstrike:falconhost:query:json
isc:bind:lameserver
isc:bind:network
isc:bind:query
isc:bind:queryerror
isc:bind:transfer
mcafee:epo
mcafee:ids
modular_alerts:notable
modular_alerts:risk
purestorage:flashblade:array
purestorage:flashblade:health
purestorage:flashblade:performance
purestorage:flashblade:space
stash

DATA INPUT

We utilized Gogen to generate sample data to load into Splunk. Gogen is a program capable of reading samples and generating events to simulate various data sources for ingest, parsing, and load testing of various systems. The program also includes splunk_app_gogen, a modular input wrapper for streaming Gogen events into Splunk.

For this test, KGI wrote Gogen configuration files to simulate twenty (20) sourcetypes matching the various Splunk TAs mentioned above.

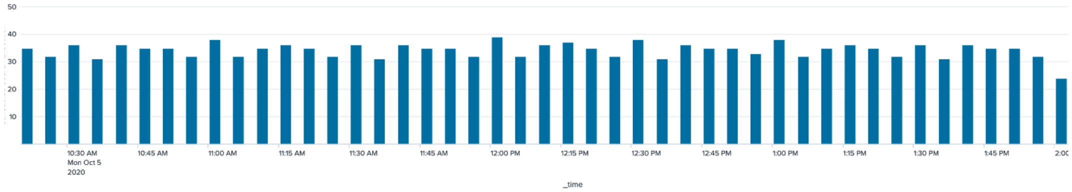
Docker containers running Splunk's official Universal Forwarder Docker image were used to send data to Splunk. These containers were also configured to install Gogen for data generation. Each container was configured to generate events equivalent to approximately 500 GB of daily ingestion. Total data ingestion was increased by altering the Gogen configuration to increase events generated. We were able to generate up to 1.5 TB of daily ingest for each universal forwarder. For tests above 1.5 TB daily ingest we increased the number of universal forwarders.

The following data ingestion rates were tested using this reference architecture:

Levels	Daily Ingest	Description
1	500+ GB (~22 GB/hr.)	Matches Splunk Reference Sizing for ES Indexers, and is a requirement to be met for any reference architecture.
2	1+ TB (~45 GB/hr.)	Double the Splunk-recommended ingest rate, representing a 50% reduction in required indexers.
3	2+ TB (~95 GB/hr.)	More than 4x the Splunk-recommended ingest rate, representing a 75% reduction in required indexers.

ENTERPRISE SECURITY TUNING

Out of the box, Splunk Enterprise Security (ES) contains many inefficiencies in the search configuration. Using KGI expertise, Enterprise Security was tuned to avoid skipped searches as much as possible while maintaining the amount of searches in the environment. This included updating scheduled search run time.



As a reminder, this tuning would be required in any large Splunk ES environment to avoid skipped searches. While results were seen in the testing environment, the tuning could be continued for additional gains in scheduled search capacity within the environment.

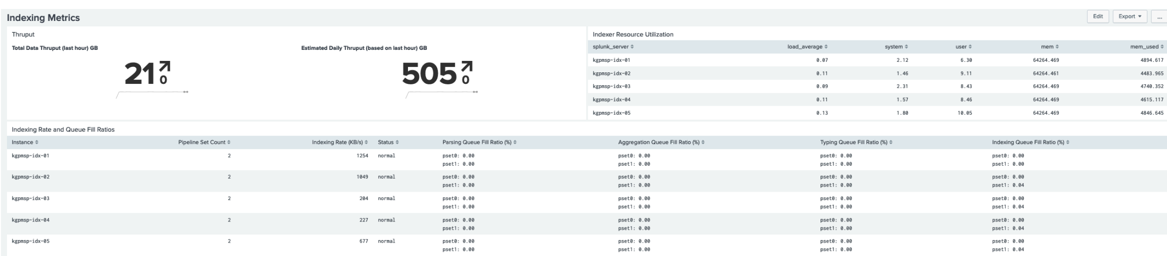
Performance Tests

DATA INGESTION

Data ingestion was set up at the following levels using Gogen: ~500 GB per day, ~1 TB per day, and ~2 TB per day. Once we reached 2 TB per day, we attempted to index as much data as possible from the universal forwarders before indexing queues stayed above 50% for sustained periods of time, or skipped searches were observed. In our testing we were able to scale the data volume up to 4 TB per day before queues began to fill up.

500 GB/day Results

Using one universal forwarder and data ingestion averaging over 500 GB/day (~22 GB/hr.), we see that the indexers remain stable and ingestion queues are not maintaining high values (throughout testing, we noticed some momentary volume in ingestion queues, but these values are not maintained unless otherwise noted).



Indexing Metrics

Thruput

Total Data Thruput (last hour) GB: 21 ↗ 0

Estimated Daily Thruput (based on last hour) GB: 505 ↗ 0

Indexer Resource Utilization

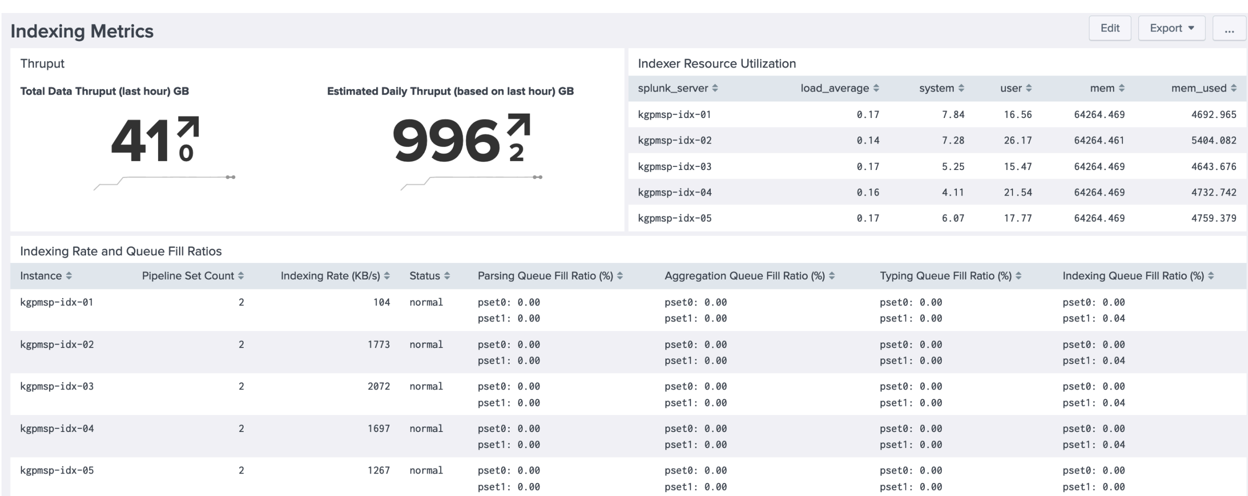
splunk_server	load_average	system	user	mem	mem_used
kgpmsp-idx-01	0.07	2.12	6.30	64264.469	4894.617
kgpmsp-idx-02	0.11	1.46	9.11	64264.461	4483.965
kgpmsp-idx-03	0.09	2.31	8.43	64264.469	4740.352
kgpmsp-idx-04	0.11	1.57	8.46	64264.469	4615.117
kgpmsp-idx-05	0.13	1.80	10.05	64264.469	4846.645

Indexing Rate and Queue Fill Ratios

Instance	Pipeline Set Count	Indexing Rate (KB/s)	Status	Parsing Queue Fill Ratio (%)	Aggregation Queue Fill Ratio (%)	Typing Queue Fill Ratio (%)	Indexing Queue Fill Ratio (%)
kgpmsp-idx-01	2	1254	normal	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00
kgpmsp-idx-02	2	1049	normal	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.04
kgpmsp-idx-03	2	204	normal	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.04
kgpmsp-idx-04	2	227	normal	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.04
kgpmsp-idx-05	2	677	normal	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.04

1 TB /day Results

When data ingestion is increased to over 1 TB /day (~45 GB/hr.) using two universal forwarders, we continue to see healthy indexers. Here we can see the queues are still not maintaining any high values.



Indexing Metrics

Thruput

Total Data Thruput (last hour) GB: 41 ↗ 0

Estimated Daily Thruput (based on last hour) GB: 996 ↗ 2

Indexer Resource Utilization

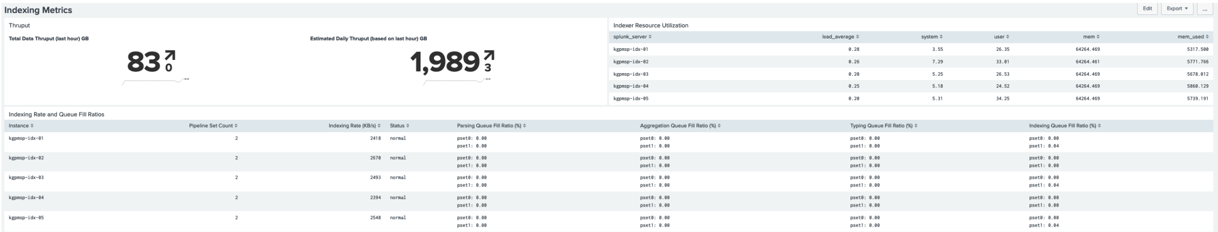
splunk_server	load_average	system	user	mem	mem_used
kgpmsp-idx-01	0.17	7.84	16.56	64264.469	4692.965
kgpmsp-idx-02	0.14	7.28	26.17	64264.461	5404.082
kgpmsp-idx-03	0.17	5.25	15.47	64264.469	4643.676
kgpmsp-idx-04	0.16	4.11	21.54	64264.469	4732.742
kgpmsp-idx-05	0.17	6.07	17.77	64264.469	4759.379

Indexing Rate and Queue Fill Ratios

Instance	Pipeline Set Count	Indexing Rate (KB/s)	Status	Parsing Queue Fill Ratio (%)	Aggregation Queue Fill Ratio (%)	Typing Queue Fill Ratio (%)	Indexing Queue Fill Ratio (%)
kgpmsp-idx-01	2	104	normal	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.04
kgpmsp-idx-02	2	1773	normal	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.04
kgpmsp-idx-03	2	2072	normal	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.04
kgpmsp-idx-04	2	1697	normal	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.04
kgpmsp-idx-05	2	1267	normal	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00

2 TB/day Results

When data ingestion is increased to 2 TB /day – four times the Splunk recommended limit – we begin to see indexer health reduced. Data ingestion queues begin to maintain high values, and the potential for data loss is introduced to the environment.



Indexing Metrics

Thruput

Total Data Thruput (last hour) GB: 83 (0)

Estimated Daily Thruput (based on last hour) GB: 1,989 (3)

Indexer Resource Utilization

splunk_server	load_average	system	user	mem	mem_used
kgpmsp-idx-01	0.28	3.55	26.35	64264.469	5317.500
kgpmsp-idx-02	0.26	7.29	33.01	64264.461	5771.766
kgpmsp-idx-03	0.20	5.25	26.53	64264.469	5678.012
kgpmsp-idx-04	0.25	5.18	24.52	64264.469	5860.129
kgpmsp-idx-05	0.20	5.31	34.25	64264.469	5739.191

Indexing Rate and Queue Fill Ratios

Instance	Pipeline Set Count	Indexing Rate (KB/s)	Status	Parsing Queue Fill Ratio (%)	Aggregation Queue Fill Ratio (%)	Typing Queue Fill Ratio (%)	Indexing Queue Fill Ratio (%)
kgpmsp-idx-01	2	2418	normal	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.04
kgpmsp-idx-02	2	2670	normal	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00
kgpmsp-idx-03	2	2493	normal	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.04
kgpmsp-idx-04	2	2394	normal	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00
kgpmsp-idx-05	2	2548	normal	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.00	pset0: 0.00 pset1: 0.04

Beyond 2TB?…

For our final test, we wanted to push the environment as far as we could with data ingestion. Ingestion was increased to over 4 TB/day (192 GB/hr.) using 8 universal forwarders. While this level of data ingest appears possible, the results are inconsistent and introduce a variety performance issues. Beyond 2TB our recommendation would be to increase the number of indexers.

SEARCH PERFORMANCE

In addition to scheduling correlation searches. The following Splunk data models were accelerated, to represent the common customer use-cases for Enterprise Security: Authentication, Change, Domain Analysis, Email, Endpoint, Intrusion Detection, Malware, Network Resolution (DNS), Network Sessions, Network Traffic, Performance, Risk Analysis, Ticket Management, Updates, Vulnerabilities, and Web.

From within Splunk, we can run a search to find the average search execution latency. According to Splunk best practices, this should never be higher than 3 ms.

Below are the results seen throughout testing:

From these results, we can determine that search latency will not be a problem even at the highest levels of data throughput.

STORAGE PERFORMANCE

Performance metrics were pulled directly from the Pure Storage Array. Below are the metrics achieved at the highest level of data ingest, at just over 2TB per day.

As we can see from the results, even at the highest data volumes, disk latency never exceeded 1.5ms, and IOPs remained relatively flat.

Conclusions

REFERENCE ARCHITECTURE INDEXER SCALING

By utilizing KGI engineering expertise, Intel white label, Pure Storage, and VMware virtualization, a sizing capability of 400 GB per day per indexer of daily ingest is achievable without negative impact on indexing or search performance. We determined this to be a safe, reliable sizing utilizing the solution through testing on 5 indexers at 95 GB of data ingestion per hour, and providing a conservative 10% margin for data surge. As such, the following scaling table is achievable with this architecture. This represents a 75% reduction in the number of indexers required for the workload. Utilizing these values, we can extrapolate the required number of indexers using the following function:

```
# of Indexers = Total Daily Ingest Volume / 400 GB
```

Utilizing this equation, the below table was created to size the number of indexers required for 3 different levels of total daily ingest.

Expected Total Daily Volume	Number of Indexers Recommended
2 TB	5
10 TB	25
20 TB	50

Appendix 1: Configuration Items

SMARTSTORE CONFIGURATION

The following details the configuration used for Splunk SmartStore on the index cluster. Some items may need to be adjusted for customer environments.

Volumes

The following volume configuration must be deployed to the indexers via the cluster master. The primary volume will be used for SmartStore cache on the indexers. The path should be set to the local path used for hot data. The remote volume is used to point the indexers to the remote S3-compatible storage provider. The remote endpoint address, access key, secret key for the remote volume should be supplied by the Pure Storage Admin.

```
[volume:remote]
storageType = remote
path = s3://pk-smartstore
remote.s3.access_key = <access_key>
remote.s3.secret_key = <secret_key>
remote.s3.endpoint = http://<remote_ip>
```

Indexes

All indexes should be configured following the below template. This should be deployed to the indexers via the cluster master. The home path should be set to the primary volume where the SmartStore cache will be located. The remote path should be set to the remote volume. The cold path and thawed path must be set to avoid errors in Splunk, although they will not be used with SmartStore. The Splunk variable $_index_name is used for easy replication of individual index configuration, but cannot be used for the thawed path definition.

```
[customIndex]
homePath = $SPLUNK_DB/$_index_name/db
remotePath = volume:remote/$_index_name
coldPath = $SPLUNK_DB/$_index_name/colddb
thawedPath = $SPLUNK_DB/customIndex/thaweddb
```

SmartStore Cache

Splunk SmartStore utilizes a cache management process that controls bucket evictions and downloads. The space utilized for cache management is configured in server.conf on the indexers. Larger cache sizes provide more performance for historical searches but require more local drive space allocated for each virtual machine. For our testing, we used the settings below on each indexer, resulting in 150 GB of cache space on each indexer dedicated to SmartStore cache. In practice, it would be beneficial to increase this cache size to meet your organizational requirements.

```
[cachemanager]
max_cache_size = 150000
```

Pure Storage, Inc., 650 Castro Street, Suite 400, Mountain View, CA 94041

1-02017 Pure Reference Design