DOWNLOAD PDF CLUSTERS FOR HIGH AVAILABILITY

Chapter 1 : High availability clustering | HA architecture | MuleSoft

High-availability clusters (also known as HA clusters or fail-over clusters) are groups of computers that support server applications that can be reliably utilized with a minimum amount of down-time.

Oracle Security Features prevent unauthorized access and changes. Data Recovery Advisor provides intelligent advise and repair of different data failures Dynamic Resource Provisioning allows for dynamic system changes. Online Patching allows for dynamic database patches of typical diagnostic patches Oracle Secure Backup provides a centralized tape backup management solution. The servers on which you want to run Oracle Clusterware must be running the same operating system. Many high availability architectures today use clusters alone to provide some rudimentary node redundancy and automatic node failover. However, when you use Oracle Clusterware, there is no need or advantage to using third-party clusterware. Oracle Clusterware provides a number of benefits over third-party clusterware: Oracle Clusterware enables you to use an entire software solution from Oracle, avoiding the cost and complexity of maintaining additional cluster software. By reducing the number of combinations of software necessary to coordinate and support, you can increase the manageability and availability of your system software. With Oracle Clusterware you can provide a cold failover cluster to protect an Oracle instance from a system or server failure. The basic function of a cold failover cluster is to monitor a database instance running on a server, and if a failure is detected, to restart the instance on a spare server in the cluster. Network addresses are failed over to the backup node. Clients on the network experience a period of lockout while the failover takes place and are then served by the other database instance once the instance has started. The cold failover cluster solution with Oracle Clusterware provides these additional advantages over a basic database architecture: Providing application-specific failure detection means Oracle Clusterware can fail over not only during the obvious cases such as when the instance is down, but also in the cases when, for example, an application query is not meeting a particular service level. High availability functionality to manage third-party applications Rolling release upgrades of Oracle Clusterware The operation of an Oracle Clusterware cold failover cluster is depicted in Figure and Figure These figures show how you can use the Oracle Clusterware framework to make both the Oracle database and your custom applications highly available. Figure shows a configuration that uses Oracle Clusterware to extend the basic Oracle Database architecture and provide cold failover cluster. In the figure, the configuration is operating in normal mode in which Node 1 is the active instance connected to the Oracle Database that is servicing applications and users. Node 2 is connected to Node 1 and to the Oracle Database, but it is currently standby mode. In the figure, Node 2 is now the active instance connected to the Oracle Database and servicing applications and users. Node 1 is connected to Node 2 and to the Oracle Database but Node 1 is currently idle, in standby mode. To provide this transparent failover capability, Oracle Clusterware requires a virtual IP address for each node in the cluster. With Oracle Clusterware you also define an application virtual IP address so users can access the application independently of the node in the cluster where the application is running. To configure an Oracle Clusterware environment, follow the step-by-step instructions in your platform-specific Oracle Clusterware installation guide. Unlike a traditional monolithic database server that is expensive and is not flexible to changing capacity and resource demands, Oracle RAC combines the processing power of multiple interconnected computers to provide system redundancy, scalability, and high availability. The clusters that are typical of Oracle RAC environments can provide continuous service for both planned and unplanned outages. Oracle RAC builds higher levels of availability on top of the standard Oracle features. All single instance high availability features, such as the Flashback technologies and online reorganization, also apply to Oracle RAC. Applications scale in an Oracle RAC environment to meet increasing data processing demands without changing the application code. In addition, allowing maintenance operations to occur on a subset of components in the cluster while the application continues to run on the rest of the cluster can reduce planned downtime. Oracle RAC exploits the redundancy that is provided by clustering to deliver availability with n - 1 node failures in an n-node cluster. Unlike the cold cluster model where one node is completely idle, all instances and nodes can be active to scale your application. The Oracle

Database with Oracle RAC architecture provides the following benefits over a traditional monolithic database server and the cold failover cluster model: Scalability across database instances Flexibility to increase processing capacity using commodity hardware without downtime or changes to the application Ability to tolerate and quickly recover from computer and instance failures measured in seconds Rolling upgrades for system and hardware changes Rolling patch upgrades for some interim patches Fast, automatic, and intelligent connection and service relocation and failover Load balancing advisory and runtime connection load balancing Comprehensive manageability integrating database and cluster features Figure shows the Oracle Database with Oracle RAC architecture. It is possible, under certain circumstances, to build and deploy an Oracle RAC system where the nodes in the cluster are separated by greater distances. This architecture is referred to as an extended cluster. For example, for a business that has a corporate campus, the extended Oracle RAC configuration could consist of individual Oracle RAC nodes being located in separate buildings. Oracle RAC on an extended cluster provides greater availability than a local Oracle RAC cluster, but an extended cluster may not completely fulfill the disaster recovery requirements of your organization. When the two data centers are located relatively close to each other, extended clusters can provide great protection for some disasters, but not all. You should determine if both sites are likely to be affected by the same disaster. For example, if the extended cluster configuration is set up properly, it can protect against disasters such as a local power outage, an airplane crash, or server room flooding. However, an extended cluster cannot protect against comprehensive disasters such as earthquakes, hurricanes, and regional floods that affect a greater geographical area. For complete disaster recovery, use the architecture described in Section 4. The advantages to using Oracle RAC on extended clusters include: Ability to fully use all of the system resources without jeopardizing the overall failover times for instance and node failures Extremely rapid recovery if one site should fail All of the Oracle RAC benefits listed in Section 4. While an extended cluster architecture can be effective and has been successfully implemented, you should implement it only in the environments involving the distance, latency, and degree of protection recommended in this discussion. Figure shows an Oracle RAC extended cluster for a configuration that has multiple active instances on six nodes at two different locations: The public and private interconnects, and the Storage Area Network SAN are all on separate dedicated channels, with each one configured redundantly. For availability reasons, the Oracle Database is a single database that is mirrored at both of the sites. Also, to prevent a full cluster outage if either site fails, the configuration includes a third voting disk on an inexpensive, low-end standard Network File System NFS mounted device. Furthermore, the standby databases can be used for read-only access and subsequently for reader farms, for reporting purposes, and for testing and development purposes. While traditional solutions such as backup and recovery from tape, storage based remote mirroring, and database log shipping can deliver some level of high availability, Data Guard provides the most comprehensive high availability and disaster recovery solution for Oracle databases. Data Guard provides a number of advantages over traditional solutions, including the following: Fast, automatic or automated failover for data corruptions, lost writes, and database and site failures Protection against data corruptions and lost writes on the primary database Reduced downtime with Data Guard rolling upgrade capabilities Ability to offload primary database activities, such as backups, queries or reporting without sacrificing RTO and RPO Site failures do not require instance restart, storage remastering, or application reconnections Transparent to applications Effective network utilization In addition, for data resident in Oracle databases, Oracle Data Guard, with its built in zero data loss capability, is more efficient, less expensive and better optimized for data protection and disaster recovery than traditional remote mirroring solutions. Oracle Data Guard provides a compelling set of technical and business reasons that justify its adoption as the disaster recovery and data protection technology of choice, over traditional remote mirroring solutions. The following list summarizes the advantages of using Oracle Data Guard compared to using remote mirroring solutions: However, if a remote mirroring solution is used for data protection, typically you must mirror the database files, the online redo logs, the archived redo logs and the control file. If the flash recovery area is on the source volume that is remotely mirrored, then you must also remotely mirror the flashback logs. Thus, compared to Data Guard, a remote mirroring solution must transmit each change many more times to the remote site. Data Guard is designed so that it does not affect the Oracle database writer

DBWR process that writes to data files, because anything that slows down DBWR process affects database performance. Compared to mirroring, Data Guard provides better performance and is more efficient, Data Guard always verifies the state of the standby database and validates the data before applying redo, and Data Guard enables you to use the standby database for updates while it continues to protect the primary database. Better suited for WANsâ€”Remote mirroring solutions based on storage systems often have a distance limitation due to the underlying communication technology Fibre Channel, ESCON used by the storage systems. In a typical example, the maximum distance between these two boxes connected in a point-to-point fashion and running synchronously can be only 10 km. Using specialized devices this distance can be extended to 66 km. However, when the standby data center is more than 66 km apart, you must use a series of repeaters and converters from third-party vendors. Better resilience and data protectionâ€”Oracle Data Guard ensures much better data protection and data resilience than remote mirroring solutions, because corruptions introduced on the production database probably can be mirrored by remote mirroring solutions to the standby site, but corruptions are eliminated by Data Guard. For example, if a stray write occurs to a disk, or there is a corruption in the file system, or the Host Bus Adaptor corrupts a block as it is written to disk, then a remote mirroring solution may propagate this corruption to the DR site. Because Data Guard only propagates the redo data in the logs, and the log file consistency is checked before it is applied, all such external corruptions are eliminated by Data Guard. Higher Flexibilityâ€”Data Guard is implemented on top of pure commodity hardware. There is no fancy or expensive hardware required. It also allows the storage to be laid out in a different fashion from the primary. For example, you can put the files on different disks, volumes, file systems, and so on. Better Functionalityâ€”Data Guard, with its full suite of data protection features Redo Apply for physical standby databases and SQL Apply for logical standby databases, multiple protection modes, push-button automated switchover and failover capabilities, automatic gap detection and resolution, GUI-driven management and monitoring framework, cascaded redo log destinations , is a much more comprehensive and effective solution optimized for data protection and disaster recovery than remote mirroring solutions. Higher ROIâ€”Businesses have to ensure that they are getting as much value as possible from their IT investments, and no IT infrastructure is sitting idle. Data Guard is designed to allow businesses get something useful out of their expensive investment in a disaster-recovery site. Typically, this is not possible with remote mirroring solutions. The recommended high availability and disaster-recovery architectures that leverage Oracle Data Guard are described in the following sections:

Chapter 2 : Hyper-V Cluster High Availability Features Design and Backups - Virtualization Howto

Clusters for High Availability: A Primer of HP Solutions introduces readers to computer system faults and to techniques for designing systems for maximum reliability and fault tolerance. The book emphasizes Hewlett-Packard's line of High Availability (HA) products, but also explains HA problems and solutions so as to provide value to all readers, regardless of whether they're running HP gear.

You can configure high availability for only the master nodes, only the proxy nodes, or for both types of node. To reduce the infrastructure requirements of your cluster, you can assign both master and proxy roles to the high availability nodes. For the master nodes, the virtual IP has to be in the same subnet. To reduce performance risk, configure more than one proxy node and 3 or 5 master nodes. You must set up shared storage across your master nodes. The file system must be accessible by your master nodes. The following directories must be mounted on this shared storage: This shared images directory is needed so that these images are kept synchronized across all master nodes. Cluster requirements These requirements are for cluster high availability only. Proxy high availability is not affected by cluster size. For cluster or master high availability, you need 3, 5 or 7 master nodes. This number of master nodes is needed to ensure fault tolerance in your cluster. You must aim for a fault tolerance of 1 or more. You must have an odd number of masters in your cluster. Having an odd master size does not change the numbers needed for majority. Majority is the number of master nodes needed for the cluster to be able to operate. However, adding extra master nodes provide a higher tolerance for failure. You can review how fault tolerance in a cluster is affected by even and odd sized clusters in Table 1: Fault tolerance in HA clusters. Fault tolerance in HA clusters Cluster size.

Chapter 3 : High Availability Architectures and Solutions

By Amitabh Tamhane Goals: This topic provides an overview of providing persistent storage for containers with data volumes backed by Cluster Shared Volumes (CSV), Storage Spaces Direct (S2D) and SMB Global Mapping.

Such clusters allow gateway administrators to group gateways to avoid single points of failure in accessing on-premises data resources. In that case, the service switches to the next gateway in the cluster, and so on. This article describes the steps you can take to create a high availability cluster of On-premises data gateways and shares best practices when setting them up. High availability gateway clusters require the November update to On-premises data gateway, or later. Setting up high availability clusters of gateways During the On-premises data gateway installation process, you can specify whether the gateway should be added to an existing gateway cluster. To add a gateway to an existing cluster, you must provide the Recovery key for the primary gateway instance for the cluster you want the new gateway to join. The primary gateway for the cluster must be running the gateway update from November or later. Managing a gateway cluster Once a gateway cluster consists of two or more gateways, all gateway management operations, such as adding a data source or granting administrative permissions to a gateway, apply to all gateways that are part of the cluster. When administrators use the Manage gateways menu item, found under the gear icon in the Power BI service, they see the list of registered clusters or individual gateways, but do not see the individual gateway instances that are members of the cluster. All new Scheduled Refresh requests and DirectQuery operations are automatically routed to the primary instance of a given gateway cluster. If the primary gateway instance is not online, the request is routed to another gateway instance in the cluster. Distribute requests traffic across all gateways in a cluster You can choose to allow traffic to be distributed across all gateways in a cluster. In the Manage gateways page in the Power BI service, when you click on a gateway cluster in the list on the left navigation tree, you can enable the option to "Distribute requests across all active gateways in this cluster. By default, that folder is C: You must be using PowerShell version 5 or newer for these scripts to work correctly. The PowerShell scripts let users perform the following operations: Set-ExecutionPolicy -ExecutionPolicy Unrestricted -Force Next, navigate to the On-premises data gateway installation folder in the PowerShell window, and import the necessary module using the following command: You must run this command and log in before other high availability commands can work properly. You can re-run the Login command to acquire a new token. AAD username and password provided as part of the command execution, not initial invocation Get-OnPremisesDataGatewayClusters Retrieves the list of gateway clusters for the logged in user.

DOWNLOAD PDF CLUSTERS FOR HIGH AVAILABILITY

Chapter 4 : Use clusters for high availability and ease of management - Splunk Documentation

High availability is a quality of a system or component that assures a high level of operational performance for a given period of time. Measuring Availability Availability is often expressed as a percentage indicating how much uptime is expected from a particular system or component in a given period of time, where a value of % would indicate that the system never fails.

Download topic as PDF Use clusters for high availability and ease of management You can group certain Splunk Enterprise components into clusters, so that they closely coordinate their activities. This serves two key purposes: This process is known as index replication. By maintaining multiple, identical copies of Splunk Enterprise data, the cluster prevents data loss while promoting data availability for searching. Splunk Enterprise clusters feature automatic failover from one indexer to the next. This means that, if one or more indexers fail, incoming data continues to get indexed and indexed data continues to be searchable. Besides enhancing high availability, clusters have other features that help to simplify the management of a distributed deployment. They include a capability to coordinate configuration updates easily across all indexers in the cluster. They include a built-in distributed search capability. They feature indexer discovery, which enables the set of forwarders to automatically load-balance across all indexers in the cluster. Even if high availability is not a concern in your environment, you can still take advantage of the simplified management features by deploying an indexer cluster without index replication. For guidance on implementing an indexer cluster, see "High availability deployment: The search heads share knowledge objects, apps, and all other configurations. You can run the same searches, view the same dashboards, and access the same search results from any search head in the cluster. Search head clusters provide several important benefits: As the number of users and the search load increases, you can add new search heads to the cluster. By combining a search head cluster with a third-party load balancer placed between users and the cluster, the topology can be transparent to the users. No single point of failure. The search head cluster uses a dynamic captain to manage the cluster. If the captain goes down, another search head automatically takes over management of the cluster.

Chapter 5 : What is High Availability? | DigitalOcean

Compare SQL Server HA Cluster Options For SQL Server high availability solutions and cluster protection in Windows environments, you have several options. SIOS DataKeeper provides key benefits including: eliminating SAN as a single point of failure, improving replication efficiency, protection for applications other than SQL Server, and protection for distributed transactions and system databases.

On Linux systems, you can also deploy replicated data queue managers RDQMs , which use a quorum-based group to provide high availability. You need to be aware of the following configuration definitions: Queue manager clusters Groups of two or more queue managers on one or more computers, providing automatic interconnection, and allowing queues to be shared among them for load balancing and redundancy. The failover transfers the state data of applications from the failing computer to another computer in the cluster and re-initiates their operation there. This provides high availability of services running within the HA cluster. Multi-instance queue managers Instances of the same queue manager configured on two or more computers. By starting multiple instances, one instance becomes the active instance and the other instances become standbys. If the active instance fails, a standby instance running on a different computer automatically takes over. HA clusters and multi-instance queue managers are alternative ways of making queue managers highly available. Do not combine them by putting a multi-instance queue manager in an HA cluster. High availability replicated data queue managers HA RDQMs Instances of the same queue manager configured on each node in a group of three Linux servers. One of the three instances is the active instance. Data from the active queue manager is synchronously replicated to the other two instances, so one of these instances can take over in the event of some failure. Disaster recovery replicated data queue managers DR RDQMs A queue manager runs on a primary node at one site, with a secondary instance of that queue manager located on a recovery node at a different site. Data is replicated between the primary instance and the secondary instance, and if the primary node is lost for some reason, the secondary instance can be made into the primary instance and started. Both nodes must be Linux servers. The replication is controlled by DRBD. Differences between multi-instance queue managers and HA clusters Multi-instance queue managers and HA clusters are alternative ways to achieve high availability for your queue managers. Here are some points that highlight the differences between the two approaches. Multi-instance queue managers include the following features: Highly available, high performance networked storage required More complex network configuration because queue manager changes IP address when it fails over HA clusters include the following features: Additional product purchase and skills are required Disks which can be switched between the nodes of the cluster are required Configuration of HA clusters is relatively complex Failover is rather slow historically, but recent HA cluster products are improving this Unnecessary failovers can occur if there are shortcomings in the scripts that are used to monitor resources such as queue managers Relationship of HA clusters to queue manager clusters Queue manager clusters provide load balancing of messages across available instances of queue manager cluster queues. This offers higher availability than a single queue manager because, following a failure of a queue manager, messaging applications can still send messages to, and access, surviving instances of a queue manager cluster queue. However, although queue manager clusters automatically route new messages to the available queue managers in a cluster, messages currently queued on an unavailable queue manager are not available until that queue manager is restarted. For this reason, queue manager clusters alone do not provide high availability of all message data or provide automatic detection of queue manager failure and automatic triggering of queue manager restart or failover. High Availability HA clusters provide these features. The two types of cluster can be used together to good effect. For an introduction to queue manager clusters, see Designing clusters.

Chapter 6 : What is a High Availability Cluster (HA Cluster)? - Definition from Techopedia

High Availability (HA) clusters provide these features. The two types of cluster can be used together to good effect. For an introduction to queue manager clusters, see Designing clusters.

The formal engineering basis of cluster computing as a means of doing parallel work of any sort was arguably invented by Gene Amdahl of IBM , who in published what has come to be regarded as the seminal paper on parallel processing: The history of early computer clusters is more or less directly tied into the history of early networks, as one of the primary motivations for the development of a network was to link computing resources, creating a de facto computer cluster. The first production system designed as a cluster was the Burroughs B in the mids. This allowed up to four computers, each with either one or two processors, to be tightly coupled to a common disk storage subsystem in order to distribute the workload. Unlike standard multiprocessor systems, each computer could be restarted without disrupting overall operation. The ARC and VAXcluster products not only supported parallel computing, but also shared file systems and peripheral devices. The idea was to provide the advantages of parallel processing, while maintaining data reliability and uniqueness. Within the same time frame, while computer clusters used parallelism outside the computer on a commodity network, supercomputers began to use them within the same computer. Following the success of the CDC in , the Cray 1 was delivered in , and introduced internal parallelism via vector processing. Attributes of clusters[edit] A load balancing cluster with two servers and N user stations Galician. Computer clusters may be configured for different purposes ranging from general purpose business needs such as web-service support, to computation-intensive scientific calculations. In either case, the cluster may use a high-availability approach. Note that the attributes described below are not exclusive and a "computer cluster" may also use a high-availability approach, etc. For example, a web server cluster may assign different queries to different nodes, so the overall response time will be optimized. Very tightly coupled computer clusters are designed for work that may approach " supercomputing ". They operate by having redundant nodes , which are then used to provide service when system components fail. HA cluster implementations attempt to use redundancy of cluster components to eliminate single points of failure. There are commercial implementations of High-Availability clusters for many operating systems. Benefits[edit] Clusters are primarily designed with performance in mind, but installations are based on many other factors. Fault tolerance the ability for a system to continue working with a malfunctioning node allows for scalability, and in high performance situations, low frequency of maintenance routines, resource consolidation[clarification needed], and centralized management. Advantages include enabling data recovery in the event of a disaster and providing parallel data processing and high processing capacity. This means that more computers may be added to the cluster, to improve its performance, redundancy and fault tolerance. This can be an inexpensive solution for a higher performing cluster compared to scaling up a single node in the cluster. This property of computer clusters can allow for larger computational loads to be executed by a larger number of lower performing computers. When adding a new node to a cluster, reliability increase because the entire cluster does not need to be taken down. A single node can be taken down for maintenance, while the rest of the cluster takes on the load of that individual node. If you have a large number of computers clustered together, this lends itself to the use of distributed file systems and RAID , both of which can increase the reliability, and speed of a cluster. Design and configuration[edit] A typical Beowulf configuration. One of the issues in designing a cluster is how tightly coupled the individual nodes may be. For instance, a single computer job may require frequent communication among nodes: The other extreme is where a computer job uses one or few nodes, and needs little or no inter-node communication, approaching grid computing. In a Beowulf cluster , the application programs never see the computational nodes also called slave computers but only interact with the "Master" which is a specific computer handling the scheduling and management of the slaves. However, the private slave network may also have a large and shared file server that stores global persistent data, accessed by the slaves as needed. Another example of consumer game product is the Nvidia Tesla Personal Supercomputer workstation, which uses multiple graphics accelerator processor chips. Besides game consoles, high-end

graphics cards too can be used instead. With the advent of virtualization , the cluster nodes may run on separate physical computers with different operating systems which are painted above with a virtual layer to look similar. An example implementation is Xen as the virtualization manager with Linux-HA. One of the elements that distinguished the three classes at that time was that the early supercomputers relied on shared memory. To date clusters do not typically use physically shared memory, while many supercomputer architectures have also abandoned it. However, the use of a clustered file system is essential in modern computer clusters. PVM must be directly installed on every cluster node and provides a set of software libraries that paint the node as a "parallel virtual machine". PVM provides a run-time environment for message-passing, task and resource management, and fault notification. Rather than starting anew, the design of MPI drew on various features available in commercial systems of the time. The MPI specifications then gave rise to specific implementations. In a heterogeneous CPU-GPU cluster with a complex application environment, the performance of each job depends on the characteristics of the underlying cluster. There are two classes of fencing methods; one disables a node itself, and the other disallows access to resources such as shared disks. For instance, power fencing uses a power controller to turn off an inoperable node. Software development and administration[edit] Parallel programming[edit] Load balancing clusters such as web servers use cluster architectures to support a large number of users and typically each user request is routed to a specific node, achieving task parallelism without multi-node cooperation, given that the main goal of the system is providing rapid user access to shared data. However, "computer clusters" which perform complex computations for a small number of users need to take advantage of the parallel processing capabilities of the cluster and partition "the same computation" among several nodes. Checkpointing can restore the system to a stable state so that processing can resume without having to recompute results. Linux Virtual Server , Linux-HA - director-based clusters that allow incoming requests for services to be distributed across multiple cluster nodes. Other approaches[edit] Although most computer clusters are permanent fixtures, attempts at flash mob computing have been made to build short-lived clusters for specific computations. However, larger-scale volunteer computing systems such as BOINC -based systems have had more followers.

Chapter 7 : How to Configure VMware High Availability (HA) Cluster - TechSupport

Cluster Sets is the new cloud scale-out technology that increases cluster node count in a single Software Defined Data Center (SDDC) cloud by orders of magnitude. A Cluster Set is a loosely-coupled federated grouping of multiple Failover Clusters: compute, storage or hyper-converged.

While handling increased system load is a common concern, decreasing downtime and eliminating single points of failure are just as important. High availability is a quality of infrastructure design at scale that addresses these latter considerations. What Is High Availability? In computing, the term availability is used to describe the period of time when a service is available, as well as the time required by a system to respond to a request made by a user. High availability is a quality of a system or component that assures a high level of operational performance for a given period of time. These values are calculated based on several factors, including both scheduled and unscheduled maintenance periods, as well as the time to recover from a possible system failure. How Does High Availability Work? High availability functions as a failure response mechanism for infrastructure. The way that it works is quite simple conceptually but typically requires some specialized software and configuration. When Is High Availability Important? When setting up robust production systems, minimizing downtime and service interruptions is often a high priority. Regardless of how reliable your systems and software are, problems can occur that can bring down your applications or your servers. Highly available systems can recover from server or component failure automatically. What Makes a System Highly Available? One of the goals of high availability is to eliminate single points of failure in your infrastructure. A single point of failure is a component of your technology stack that would cause a service interruption if it became unavailable. As such, any component that is a requisite for the proper functionality of your application that does not have redundancy is considered to be a single point of failure. For instance, imagine you have an infrastructure consisting of two identical, redundant web servers behind a load balancer. The traffic coming from clients will be equally distributed between the web servers, but if one of the servers goes down, the load balancer will redirect all traffic to the remaining online server. The web server layer in this scenario is not a single point of failure because: With the described scenario, which is not uncommon in real life, the load balancing layer itself remains a single point of failure. Redundancy alone cannot guarantee high availability. A mechanism must be in place for detecting failures and taking action when one of the components of your stack becomes unavailable. Failure detection and recovery for redundant systems can be implemented using a top-to-bottom approach: In our previous example scenario, the load balancer is the top layer. If one of the web servers bottom layer becomes unavailable, the load balancer will stop redirecting requests for that specific server. This approach tends to be simpler, but it has limitations: Creating a failure detection service for the load balancer in an external server would simply create a new single point of failure. With such a scenario, a distributed approach is necessary. Multiple redundant nodes must be connected together as a cluster where each node should be equally capable of failure detection and recovery. A change like this can take a considerable amount of time to be propagated on the Internet, which would cause a serious downtime to this system. A possible solution is to use DNS round-robin load balancing. However, this approach is not reliable as it leaves failover the client-side application. A more robust and reliable solution is to use systems that allow for flexible IP address remapping, such as floating IPs. On demand IP address remapping eliminates the propagation and caching issues inherent in DNS changes by providing a static IP address that can be easily remapped when needed. The domain name can remain associated with the same IP address, while the IP address itself is moved between servers. This is how a highly available infrastructure using Floating IPs looks like: There are several components that must be carefully taken into consideration for implementing high availability in practice. Much more than a software implementation, high availability depends on factors such as: Having redundant servers in different datacenters and geographical areas will increase reliability. Highly available systems must account for data safety in the event of a failure. It is important that a redundant network strategy is in place for possible failures. Each layer of a highly available system will have different needs in terms of software and configuration. However, at the application level,

load balancers represent an essential piece of software for creating any high availability setup. HAProxy High Availability Proxy is a common choice for load balancing, as it can handle load balancing at multiple layers, and for different kinds of servers, including database servers. Moving up in the system stack, it is important to implement a reliable redundant solution for your application entry point, normally the load balancer. To remove this single point of failure, as mentioned before, we need to implement a cluster of load balancers behind a Floating IP. Corosync and Pacemaker are popular choices for creating such a setup, on both Ubuntu and CentOS servers. Conclusion High availability is an important subset of reliability engineering, focused towards assuring that a system or component has a high level of operational performance in a given period of time. At a first glance, its implementation might seem quite complex; however, it can bring tremendous benefits for systems that require increased reliability.

Chapter 8 : High Availability | Cluster Administration | OKD Latest

Setting up high availability clusters of gateways. During the On-premises data gateway installation process, you can specify whether the gateway should be added to an existing gateway cluster. To add a gateway to an existing cluster, you must provide the Recovery key for the primary gateway instance for the cluster you want the new gateway to join. The primary gateway for the cluster must be running the gateway update from November or later.

Virtualization has allowed for a powerful abstraction from the underlying physical hardware that allows production workloads to be mobile. This opens up powerful advantages over running server workloads on physical baremetal. There are many mechanisms built into Hyper-V that allows organizations to have the tools needed to ensure high availability for virtual machines running in a Hyper-V environment. What exactly is high availability? How can third party tools bolster high availability with Hyper-V providing backups? What is High Availability? Events that can disrupt the up-time of these key areas include hardware failure, network failure or other outages based on load or application failure. While it is a part of the overall business continuity plan for organizations, it is not a disaster recovery mechanism, but rather a means to ensure business operations meet with the desired SLAs put in place and can withstand disruptions to infrastructure without losing availability to applications or data essential to business operations. Windows Failover Clustering Cluster Shared Volumes CSV Guest Clustering The above list of built in features and architecture available with Microsoft Hyper-V provides powerful high availability mechanisms that Hyper-V administrators can use to design highly available production systems. In a failed cluster node scenario, virtual machines that were running on the failed host, are restarted on a healthy host left in the Windows Failover Cluster. By using highly available network and storage design that is common to Windows Failover Clusters, Hyper-V administrators have a highly available set of cluster nodes that are able to house virtual machines, even if a node fails due to hardware or other reasons. This functionality is an integral part of highly available virtual machines as it allows them to be mobile without any downtime. This allows Hyper-V administrators to have proactive or planned downtime since virtual machines can be Live Migrated to take a host down. With Windows R2, cluster shared volumes were introduced. Prior to that release, Hyper-V hosts had to have multiple disks provisioned since only one host could access the storage at a time. This greatly complicated migrations. Cluster Shared Volumes in Hyper-V Cluster allow simultaneous access Cluster Shared Volumes also provide resiliency benefits when compared to other forms of storage in a Hyper-V cluster since multiple connections are made between nodes in the cluster and the CSV volume. This provides path redundancy. If one path goes down, the communication with a CSV volume can still traverse an alternate path. When you think about it, even if your virtual machine is highly available due to the HA mechanisms in place by Hyper-V on top of a Windows Failover Cluster, if a host the virtual machine lives on fails, the virtual machine will still go down and become inaccessible for a short time. It will be restarted on a healthy Hyper-V host, however, downtime is still experienced. However, when you have two or more clustered virtual machines inside your physical Windows Failover Cluster hosting Hyper-V, if one of your VMs goes down, you still can access your application as the clustered application, such as SQL Server, fails over to the other node in the application cluster. This bolsters high availability to business-critical application as it prevents any downtime. Some points to remember: Not all applications support guest clustering or clustering at all Licensing will generally be more expensive with guest clustering as you have to purchase more software licenses It requires more resources from your Hyper-V environment since more VMs are running Hyper-V Cluster High Availability and Backups Despite all the high availability mechanisms that can be put in place, including the ones we have covered above, there is still the need to provide data protection. High availability protects you from hardware, network, or application failure, but it does not protect you from accidental or intentional data loss that can happen. You still need to protect your data. Hyper-V clusters provide some interesting challenges for data protection solutions that must be able to interact with and protect your Hyper-V virtual machine data efficiently, effectively, and in a way that is consistent. Vembu BDR Suite is a tremendously good data protection solution that we have covered in various blog posts. Vembu is set to announce really exciting

functionality that will be unveiled in an upcoming webinar scheduled as relates to Hyper-V High Availability and Hyper-V Clusters. Be sure not to miss it! Date â€" Tuesday, July

Chapter 9 : High availability IBMÂ® Cloud Private clusters

Creating Highly Available Clusters with kubeadm. This page explains two different approaches to setting up a highly available Kubernetes cluster using kubeadm: With stacked masters. This approach requires less infrastructure. etcd members and control plane nodes are co-located. With an external etcd cluster. This approach requires more infrastructure.

In this article, first I will introduce the basic concepts of VMware HA then I will to show you its configuration steps. What can we do to reduce the impact of a host failing without prior notice? You want to ensure high availability of your VMs. VMware HA will restart the VMs on the remaining hosts in the cluster in case of hardware failure of one of your hosts. You want to distribute your VMs evenly across the resources of the cluster. The distributed resource scheduler DRS to be discussed later will make sure that each host runs with the same level of memory and CPU utilization. If, for any reason, the use of memory or CPU rises or falls , DRS kicks in and vMotion moves the VMs to other hosts within your cluster in order to guarantee an equal level of utilization of resources across the cluster. In other words, HA will protect your VMs from host hardware failure, while DRS ensures that utilization of resources across the cluster is equal. So far, so simple. Shared vs hyper-converged storage VMware HA works well in traditional architectures with shared storage, but what about hyper-converged storage? All hosts present in the cluster can access shared storage and run VMs from a shared data store. Such architecture has been commonplace for about 15 years and has proven reliable and efficient. So basically, after you hit that limit, you end up creating another silo where you can put some hosts together with new shared storage. Recently, hyper-converged architectures have become popular and are available from different vendors including VMware with VSAN , where shared storage devices are replaced with the local disks in the hosts of the cluster. These local disks are pooled together across the cluster in order to create a single virtual shared data store that is visible to all hosts. This is a software-only solution that can leverage high-speed flash devices, optionally in combination with rotating media. This tells us that even very small enterprises can benefit from this easy-to-use technology. HA configuration options VMware HA is configurable through an assistant, allowing you to specify several options. Host Monitoring â€" You would enable this to allow hosts in the cluster to exchange network heartbeats and to allow vSphere HA to take action when it detects failures. Note that host monitoring is required for the vSphere Fault Tolerance FT recovery process to work properly. FT is another advanced, cool technology that allows you to protect your workflows in case of hardware failure. However, compared to HA, it does that in real time, without downtime and without the need for a VM restart! If you enable it, you have to choose a policy of how it is enforced. Admission control will prevent the starting of VMs if the cluster does not have sufficient resources memory or CPU. Virtual Machine Options â€" What happens when a failure occurs? The VM options allow you to set the VM restart priority and the host isolation response. Datastore Heartbeating â€" You have the possibility to check a secondary communication channel so vSphere can verify that a host is down. In this option, the heartbeats travel through a data store or several. VMware datastore heartbeating provides an additional option for determining whether a host is in a failed state. The datastore heartbeat function helps greatly in determining the difference between a host which has failed and one that has merely been isolated from others. This packaged offer can be purchased online or through software resellers. The latest version of vSphere is 6. Two network switches with ports. By having a dedicated storage network, you can avoid VLANs. Network configuration The first thing to do when preparing for VMware vSphere deployment is to plan ahead in your network, not only in the network-only environment but also in the storage environment, as most storage-based solutions are Ethernet-based iSCSI or NFS. It is a small network configuration that I will place emphasis on today. The guide will work with the smallest possible redundant storage network configuration for two hosts connected to iSCSI storage. There are only two hosts in the image, but you can easily add a third and, potentially, a fourth host in order to expand your cluster. This merely depends on the requirements for properly sizing the network switches with a correct number of network ports, in order to satisfy future growth. All IP addresses of Network 1, marked in blue, will use a base address of the subnet Then, we do the same

thing for Network 2, which has a base address of In the image, this network is marked in red. As you can see, all the ESXi port groups are following this numbering scheme to stay within the same range. I am aware that my guide represents a very generous storage array with four network ports per storage processor. As we now have the networking hardware for our HA cluster in place, we can start by configuring vSphere. Open the vSphere client and start the assistant: Now, you just have to follow the Add Network Wizard. Create a new vSphere standard switch â€" Network label In the next step, you have to assign an IP address. Today we learned how to: VMware uses Storage Array Type Plug-Ins SATPs , which run in conjunction with the VMware NMP and are responsible for array-specific operations, such as monitoring the health of each physical path, reporting physical path changes, or activating passive paths for active-passive arrays. We have created two separate networks between each host and storage array, which gives us a backup in case we have any of the following issues: The vSwitch had been configured with 2 NICs. Now we need to accomplish four steps: Enable the iSCSI initiator. After the adapter has been created, select the software iSCSI adapter in the list, right-click, and select Properties in order to configure it. The iSCSI adapter is usually created as vmhba Select the second tab from the left: Repeat for the iscsi Configure the iSCSI target. Now that we have added our configuration, we need to point this initiator to our array. Add all IP addresses of the array. Create a shared datastore. You should see a new datastore appear. Shared datastore created Step 4: Set up a Round Robin storage policy. This policy will use both paths randomly and spread availability and VM storage traffic across both links as we have two NICs on our storage network. Specifying Round Robin as default path We should see now multiple paths to the storage: In the next and final step of this article, I will cover the HA configuration. This will allow our VMs to be restarted in case of unwanted hardware failure, such as a host that loses a CPU or motherboard or simply goes blue screen because of a faulty RAM. After activating HA, our cluster will become fully resilient in case we have a hardware failure on one of our hosts. A minimum of two hosts in the cluster â€" You can have up to 64 hosts in vSphere cluster. Shared Storage â€" Every server that is part of the HA cluster needs to have access to at least one shared storage. Network â€" You must connect all hosts to at least one management network. Licensing â€" All hosts must have licenses for HA. You need at least vSphere Essentials. VMware tools â€" It is highly recommended that you install VMware tools on all hosts, and it is required if you want to work with the VM monitoring feature. This feature does not appear in the vSphere client. However, in case the vCenter server becomes unavailable, those functions continue to work. Create a Datacenter Object The datacenter object is at the top-level object. Bellow you will have clusters, individual hosts, folders, VMs, etc. Now, we are ready to create the cluster. As you can see in the screenshot above, the wizard looks quite similar in both management clients. New Cluster Wizard Step 3: Turn on vSphere HA cluster vSphere HA cluster configuration options Many options exist that allow you to adjust the behavior of the cluster in case of a hardware failure in one of your hosts. Only the vSphere web client offers the more advanced options: For instance, you could assign a high priority to database servers, a medium priority to file servers, and a low priority to web servers. This means that the database servers would be up and running before the file servers and webservers. VM Monitoring Datastore Heartbeating â€" In case the management network fails, Datastore Heartbeating allows you to determine if the host is in a failed state or just isolated from other hosts within the cluster. This can be a simple alarm or the restart of VM on another host. This completes my VMware HA configurations steps.