VERITAS™

Maximizing Business Continuity Success

Learnings from Best in Class Implementations

Who should read this paper

Investments in High Availability and Disaster Recovery tools to support business continuity objectives can fall short of the mark when not managed well. Indeed, the presence of such tools can lead to a false sense of security, resulting in complacency in the face of insidious challenges until it is too late. IT professionals responsible for providing highly available environments to support critical business operations will do well to understand how causes such as configuration drift come about rendering the their plans ineffective and what to do about them.

September 2015

Contents

Executive Summary

Enterprises concerned with business continuity make significant investments in High Availability (HA) and Disaster Recovery (DR) to help ensure that business-critical applications remain available. Yet, various factors combine to make achieving very high availability a challenging proposition. These factors include configuration drift (the almost inevitable change in configurations from their intended and deployed state), the IT organization-spanning nature of responsibilities (from network team, to applications to storage) for managing the entire IT stack necessary to support business continuity objectives and the proliferation of management tools for managing the diverse environment. Adopting certain best practices can help mitigate these challenges. Using an automated, environment-spanning tool to monitor the environment can help make the difference between an environment that may not withstand component failures and natural disasters and one that is well managed and quite likely to support your business continuity objectives well.

Introduction

Downtime of critical business services, associated data loss and the need to maintain required application performance levels are top concerns for all enterprises.   The cost and risks associated with downtime and lost data are significant enough to justify a major investment in High Availability (HA) and Disaster Recovery (DR) solutions. But despite the plethora of High Availability and Disaster Recovery technologies that can be found in the typical enterprise data center—clustering, load-balancing, replication, as well as new technologies such as grid computing, parallel clusters, and virtualization-based high availability—downtime and data loss are still quite common. The investment in these solutions is not being reflected in a commensurate reduction in risk. This is not to say that investing in traditional High Availability and Disaster Recovery solutions is not recommended, but rather an indication that something critical in how these solutions are often being managed is still missing.

Challenges to High Availability Environments

Three factors in particular, contribute to making management of HA/ DR environments challenging and serve to raise the risk of an outage or data loss:

- Configuration drift
- Need for cross-domain and cross-vendor integration
- Proliferation of management tools

Configuration Drift

Changes to an IT environment occur frequently as part of normal operations – these include: operating systems, patches, and software installs or updates; storage allocations changes; kernel, system and networking parameters adjustments; hardware configurations (server, network, SAN) updates; etc. Each time a change is made, the IT professional making the change must consider if there are any continuity implications for the IT environment. In many cases there are and action needs to be taken to keep the environments in synch.

This fact, in itself, introduces the risk that some required changes may be left out. It's extremely difficult to notice such discrepancies, especially when multiple teams, such as storage, server, network and DBA, must all take part. But even if the change control processes are perfect, once an update has been made to all components (e.g., all nodes of the same cluster), there's another frustrating asymmetry that must be faced: many HA solutions involve both passive and active components. Even when so-called "active-active" configurations are used, often it is only a means to improve utilization. For example, application A runs on node 1. Node 2 is a standby that does not concurrently run application A. To better utilize node 2, it is decided to let it run application B in the meanwhile. And here's the asymmetry: we can tell if the active components work, simply because they are in use; but what about the standbys? How can we know they are ready to take over when needed? Consider, for example, a cluster standby missing SAN paths to a shared storage volume, or one missing a correct startup parameter. This may not be detected unless the failover process is actively tested. However, failover testing does not happen very frequently meaning hidden vulnerabilities may linger for weeks or months and these undetected hidden risks can lead to failures.

Need for Cross-Domain and Cross-Vendor Integration

An HA environment typically spans a range of components (see Figure 1), such as networks, servers and storage and responsibility for configuring and managing these components typically also correspond to separate organizational teams.

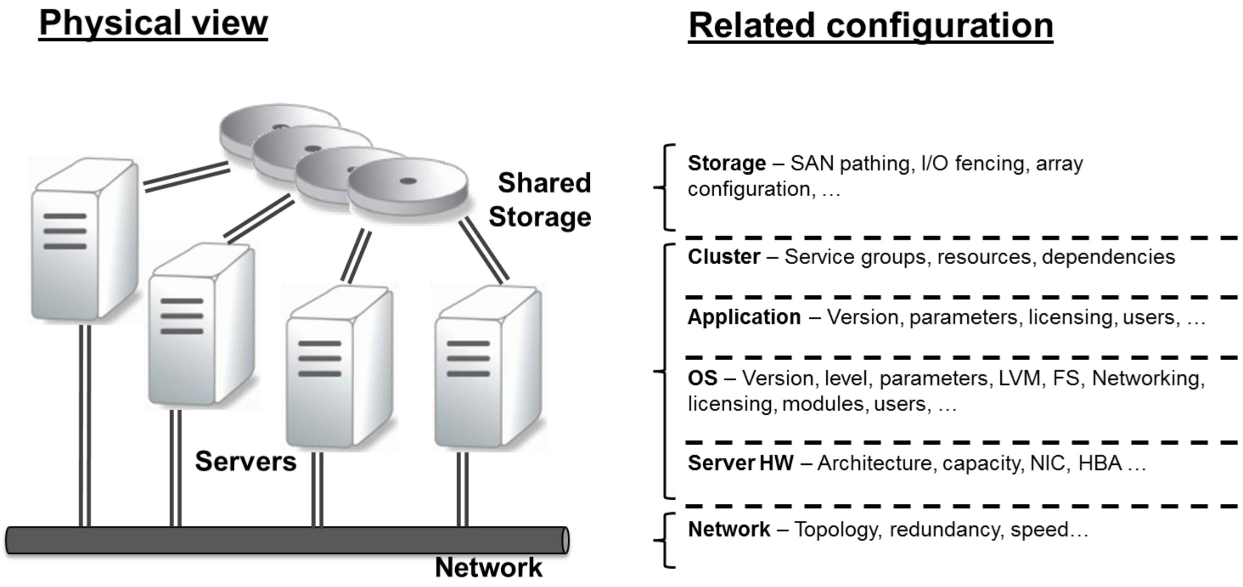
Physical view
Related configuration
Shared Storage
Servers
Network
Storage – SAN pathing, I/O fencing, array configuration, …
Cluster – Service groups, resources, dependencies
Application – Version, parameters, licensing, users, …
OS – Version, level, parameters, LVM, FS, Networking, licensing, modules, users, …
Server HW – Architecture, capacity, NIC, HBA …
Network – Topology, redundancy, speed…

Figure 1 – Components of an HA environment

Often, more than one subject matter expert is required to correctly configure the relevant layers. Miscommunications may result in hidden discrepancies. For example, a Database Administrator (DBA) may wish to eliminate any single-point-of-failure in a mission-critical database, and hence, configure redundant database control files, taking care to place each copy on a different file system. However, it may be that all those file systems actually reside on the same physical SAN volumes, a fact the DBA could not readily identify.

Another important aspect adding to the complexity is the need to use hardware and software from multiple vendors (storage, server, OS, cluster software, multi-pathing, etc.). Vendors usually publish specific guidelines and best practices describing configuration of components, minimum required settings, etc. In general, it is a good idea to follow these vendor-specified best practices for deploying their products. Failure to do so can result in sub-optimal configurations and an increase in risk to continuity.

Proliferation of Management Tools

Given the diversity of vendors, there is no standard tool kit for managing HA configurations in a consistent manner to help avoid configuration drift. Instead, IT administrators must use multiple point-solution tools such as storage resource management tools, cluster management consoles, network management tools, server provisioning tools, and other newer virtualization consoles (e.g., vCenter and SRM in VMware environments) to manage their environments.

Configuration Drift – Deep Dive

Configuration drift can arise through changes in the various layers of the IT environment. While they all arise as a result of a range of actions, they share a common end result – a potential increase in risk and likelihood of outages and data loss.

Cluster Related

Even when thoroughly tested and validated, clusters may become unstable as a result of maintenance-introduced configuration drift that might result in either unplanned downtime or data loss.

Examples of most frequently encountered risks include:

- **Storage access** – As more shared storage volumes are added, there is a chance that some of the standbys will not have correct SAN access. It is extremely difficult to notice such discrepancies, especially since standby nodes usually refrain from using shared storage resources until they need to take over. Even if all devices are accessible, you must verify that standby nodes have the same number of configured SAN I/O paths as well as the same redundancy setting. It is also advisable to monitor path availability. Poor performance after a failover is often associated with a standby having fewer available I/O paths than the formerly active node.
- **Degraded mode / bad state** – Clusters usually keep track of their state, as it may change from the optimum. Some state changes could be the result of an unnoticed degradation in some of the components (e.g., faulty or "noisy" Ethernet adapters). Others could be the result of incomplete maintenance activity (e.g., standby was suspended, or "frozen" to allow an upgrade, but was not brought back to normal state). Correctly monitoring your cluster state could save hours of unplanned downtime.
- **Incorrect resource configuration** – Sometimes clusters have no easy way of identifying whether a configuration file/repository on one of the nodes is different than the others. Make sure to compare them periodically.
- **Cluster resource to physical resource mismatch** – It is not uncommon to find cluster resources that point to a non-existent physical resource such as a missing mount-point (for UNIX) or the wrong drive letter (for Windows), the wrong Ethernet adapter, etc.

- **Storage control devices** – This is an often overlooked area. Many storage arrays (for example those from EMC or Hitachi Data Systems) allow a standby to take over shared devices only if it can communicate with the array. Usually, you need to present a host with at least one storage control device per concurrent operation. If you have too few control devices assigned to a host, it might hang during an attempt to take over multiple resources. Consult your cluster admin guide to determine the right number of devices to configure.

Application Related

Sometimes clusters do not share application binaries and configuration files. When these are updated, you must remember to perform the same maintenance operations on all cluster nodes. Failure to do so can lead to configuration drift. It is a good practice to periodically audit your configuration to verify that you have:

- Same installed versions /patches
- Identical or compatible configuration files
- License information
- Network objects: listening port, listener configuration
- All application data on shared storage. (In geo-clusters, make sure it is on replicated storage, as some storage resources are private to a local cluster and others are global.)

Operating System and Hardware Related

Periodically verify that your cluster nodes have similar configuration. In large environments differences can appear across a range of aspects:

- Hardware resources (memory, CPU)
- OS version, patches, licensing
- Installed system utilities, such as LVM, multi-pathing, storage and HBA utilities
- Kernel configuration (e.g., you have increased I/O queue depth for number of max processes on some nodes, but not on all)
- Network services (e.g., time, DNS) - In geo-cluster or metro-cluster configurations make sure that nodes point to a local service (e.g., to a local DNS server rather than to the same DNS server as nodes on the other site)

- Networked storage – Make sure that critical mounted networked file systems are accessible by all nodes (with the same options, mode, permission, protocol version, etc.). Pay attention to geo or campus clusters, as you should keep each node pointing at local resources
- HBA and multi-pathing configuration differences

Network Related

- Link / team speed, mode (e.g., 2 teamed 1Gbps Ethernet adapters in one node vs. 1 100Mbps link in another)
- Hidden single point of failure (e.g., both private or public links on the same switch / VLAN)
- Low level / low latency stack issues (e.g., LLC, serial heartbeat):
- Misconfiguration
- Some are non-routable
- Firewall configuration – Make sure internal firewalls have the same ports allowed. External firewalls should allow same access rights to all cluster nodes

Storage and SAN Related

- Missing SAN access to shared devices as a result of zoning or masking misconfiguration
- Non-redundant SAN I/O paths
- SAN security – A non-cluster member that has access to a shared storage device. This is a relatively vulnerable spot. The storage team should periodically verify that only valid cluster nodes can access cluster volumes
- Replication issues in geo-clusters – Make sure all data is replicated. If you are using more than one storage volume, make sure all volumes are on the same storage consistency group.
- Mixed storage tiers – It is highly recommended to make sure all shared storage devices are based on the same storage architecture and tier

Lessons Learned from Best-in-Class High Availability Implementations

Lesson #1: Encourage Standardization

Minimize the number of possible HA configurations, and strive to standardize and re-use the same design patterns, standardizing on the same clustering technology, using the same software versions and patch-levels on all clusters to the extent practical.

When possible, it is also recommended to use the same, internally certified, “golden image” to template cluster nodes.

Finally, it is important to document and publish standards to facilitate consistency of future HA systems. Some important areas to include are:

- Minimum hardware requirements (power, internal disks, NICs, HBAs and ports)
- Networking standards (e.g., private vs. public network requirements, proprietary low-latency protocol configuration, firewall requirements)
- Software requirements (e.g., cluster software, multi-pathing software, custom storage agents and CLIs, runtime frameworks, etc.). Try to specify exact versions
- Storage requirements (e.g., multi-pathing, zoning and masking guidelines, control device requirements and best practices)
- Naming convention (for nodes, virtual IPs, services, etc.)

Lesson #2: Develop a Collaborative Culture

A successful HA environment requires correct configuration and management of network, storage, server and often database as well. Without adopting a cross-domain culture and getting all relevant teams educated and engaged, sub-optimal or even incorrect configurations might be reached.

We recommend forming an HA team (dedicated or virtual), that will:

- Include members from all relevant teams
- Make sure all teams have a high degree of education on HA principles and technical requirements
- Jointly design and periodically review HA architectures and configurations
- Jointly define auditing and testing goals

Many organizations new to this concept are skeptical at first, and their reaction is that this approach is ineffective, or a waste of time. In reality, once geared up, there is very little overhead. A one-hour team review each month usually suffices, except when new designs or architecture refresh processes are in motion. The payoffs far outweigh the time invested. Better communication and increased awareness translates to more efficient deployments and dramatically reduces the time required to resolve issues, should they occur.

Lesson #3: Conduct Frequent Fail-over Testing and Auditing

Testing and auditing your HA configurations are important factors in ensuring successful recovery.

The most effective approach requires rotating your active nodes regularly and frequently (e.g., a very aggressive approach would require failing over to a different node each weekend, and letting it run in production the following week). However, such an aggressive approach is inappropriate in most environments. Most view this approach as impractical due to one of the following two reasons:

1. Fail-over is still risky and involves downtime, and therefore requires business approval, which can simply not be granted that frequently.
2. Production and standby systems are not always fully symmetrical. For example:
   - Standbys have less capacity, so you cannot afford to let them run your production applications for the entire week
   - Standbys are located in sites with sub-optimal network (bandwidth, response times)
   - Standbys are installed with less critical applications (e.g., development or testing) that cannot be also installed on the primaries, rendering server rotation impractical

Whereas frequent fail-overs may not be practical, it is practical to audit frequently to “keep on top of” the situation.

Lesson #4: Automate Auditing

While testing and server rotation can be expensive, disruptive and not always practical, automated auditing represents the most successful and proactive approach. This involves either using a commercial, off-the-shelf tool or a set of customized home-grown scripts. Guidelines for successful auditing include:

- As a minimum, automate the collection of relevant configuration items (hardware, OS and software configuration, storage allocation, cluster configuration, networking configuration, etc.). Automatic data collection can dramatically reduce the time and effort involved in testing, auditing, and preparing for future downtime. Without regularly collected configuration data, it is almost impossible to perform post-mortem analysis when actual downtime does occur.
- The next level, which could prove more difficult to reach unless dedicated tools are used, is to automatically search for known vulnerabilities, such as those described in earlier sections. Automated tests which are non-intrusive in nature can be run frequently to help minimize configuration drifts and associated risks.

Supporting Business Continuity with Veritas™ Risk Advisor

Automated auditing is a key driver for improving the readiness of your HA environment and reducing business continuity risks. While writing your own scripts may seem like an attractive approach, it is often difficult and limited because:

- It requires writing and debugging a large number of scripts (some relatively complex) based on an understanding of management frameworks for a diverse set of components
- You need to make sure you configure and run the scripts on all relevant hosts (existing and new)
- You are limited to what your own experience teaches you
- Personnel changes can render the most skillfully designed scripts impossible to maintain

Risk Advisor offers an alternative approach which can prove more cost-effective and much more comprehensive than homegrown solutions.

Risk Advisor employs an agent-less technology that runs on a single dedicated server. Setting up Risk Advisor is simple, and can be accomplished in less than a day. You can configure Risk Advisor to scan your environment for High Availability and Disaster Recovery vulnerabilities frequently (every day of the week if you wish), allowing you to:

- Automatically discover your servers, clusters, storage arrays, SAN configuration, replication configuration and database configuration
- Have visibility into detailed configuration information for all layers in your IT stack, store the information in a central repository, track change history, and generate custom reports
- Automatically test the validity of your HA/DR configurations against a risk-detection knowledgebase containing nearly 6,000 different potential failure points (identified from experience with a number of customers) that are updated on a weekly basis
- Present and communicate the identified risks to your IT counterparts in an actionable format, including graphical diagrams of the environment at risk, a description of the root cause, and remediation instructions

The benefits delivered by Risk Advisor help ensure that your High Availability and Disaster Recovery investments will pay off:

- Providing end-to-end visibility into your IT stack, the equivalent of running a complete High Availability and/or Disaster Recovery audit
- Eliminating manual labor associated with documenting, auditing, and testing your High Availability and Disaster Recovery environment
- Minimizing downtime risk by capturing configuration drifts as they occur, providing expert advice on how to fix them, and helping keep your environment consistent and recoverable

About Veritas Technologies LLC

Veritas Technologies LLC
enables organizations to harness the power of their information, with solutions designed to serve the world's largest and most complex heterogeneous environments. Veritas works with 86 percent of Fortune 500 companies today, improving data availability and revealing insights to drive competitive advantage.

For specific country offices

and contact numbers, please

visit our website.

Veritas World Headquarters

500 East Middlefield Road

Mountain View, CA 94043

+1 (650) 933 1000

www.veritas.com