

Expect the Unexpected: Disaster Recovery for Windows Server Applications

Author: Chad Todd

Sponsored by:

neverfailTM

Expect the Unexpected: Disaster Recovery for Windows Server Applications

Look around today and you will see a world dependent on computers. Computers have become so ubiquitous that we don't even notice them anymore, but they are everywhere. It is difficult to find a business that does not rely on computers in some form. With this reliance comes a greater vulnerability and exposure to risk if employees, partners, and customers can't access the information and tools needed to maintain business operations.

The purpose of this white paper is to expose IT personnel to the different types of disasters – both large and small – facing their company. It will explain how to protect your company's profits, reputation, and productivity by minimizing your risk to disasters with high availability solutions. You will learn the difference between disaster recovery and disaster avoidance and why one is much more desirable.

This white paper can serve as a checklist when performing your own risk assessment. As you read through it make a note of the disasters to which you are vulnerable. Compare the possible methods of disaster recovery and disaster avoidance to what you are currently using. By the end of this document you will have a good understanding of your company's risks and the solutions available to minimize those risks.

The Cost of a Disaster

The first question you need to ask yourself is, "If a disaster took down our critical applications what would be the true cost to my company?" This is a two-part question. You need to look at what it would cost you to recover the failed systems and how much revenue is lost during the downtime multiplied by the number of people affected. For most companies the cost of fixing the failed systems is minimal compared to the cost of lost revenue. Table 1.1 shows the average costs of one hour of downtime for various industries.

Table 1.1 Breakdown of hourly downtime costs by industry.

Industry	Hourly Downtime Costs
Brokerage Operations	\$6,450,000
Energy	\$2,817,846
Financial Institutions	\$1,495,134
Information Technology	\$1,344,461
Insurance	\$1,202,444
Retail	\$1,107,274
Pharmaceuticals	\$1,082,252
Banking	\$996,802

Sources: IT Performance Engineering and Measurement Strategies: Quantifying Performance and Loss, Meta Group, October 2000; Fiber Channel Industry Association

There are many ways a disaster can cause you to lose revenue. You may lose the ability to receive orders from customers. You may have reduced productivity preventing you from keeping up with the demand for your product. Or, if the outage is severe enough, it may tarnish your company's reputation causing you to lose additional revenue long after your systems are back up.

Disasters Defined

Webster's dictionary defines *disaster* as "a sudden calamitous event bringing great damage, loss, or destruction." A better definition of *disaster* as it relates to the IT field is "any circumstance resulting in the unavailability of business critical information and/or systems." Disasters can cause you to lose individual files, entire file servers, critical applications, or network connectivity. Disaster can be related to outside forces or they can be self-inflicted. Some of the most common culprits include natural disasters, unexpected hazards, application failures, poor configurations, people mistakes, and hardware problems.

Natural disasters include flood, hurricane, earthquake, etc. These can be devastating to a company depending on the level of damage. Natural disasters can destroy entire datacenters. Unexpected hazards are anything out of your control that you did not expect to happen. This could be power problems such as blackouts. It could be climate issues due to inefficient or broken cooling systems in your datacenter. The number one unexpected hazard is fire. Not only do you have the danger of what the fire destroys, you also have the danger of what is damaged by putting out the fire.

Application failures occur when an application quits working. This can be due to people mistakes such as incompatible patches. It can be due to poor configurations such as configuring your Exchange server as an open relay. Sometimes applications just quit working for unknown reasons. Poor configurations happen because of a lack of knowledge by the person implementing and/or supporting your systems. These days IT departments are smaller than ever. It is not uncommon to have a few people supporting hundreds of servers and dozens of applications. It is very difficult for one person to be a master of everything under their control.

There is no definite way to protect against people mistakes. Everyone makes mistakes at some point. Since we are human we are susceptible to things that computers are not such as fatigue, bad judgment, and miscommunication, just to name a few. These vulnerabilities can cause people to create disasters without even knowing it. Some common examples are:

- Rebooting the wrong server
- Unplugging the wrong network cable from a switch
- Upgrading to an unsupported patch or service pack
- Installing incompatible software on a server

Everyone is vulnerable to hardware problems. Equipment does not last forever and eventually something will break. As long as you have a single point of failure, such as a CPU, motherboard, or power supply in your systems, you are at risk of experiencing downtime.

Protecting Against Disasters: Proactive vs. Reactive

Fighting disasters falls into two broad categories, reactive management and proactive management. Reactive management is waiting until there is a problem and then fixing it. We call this the band-aid approach. Your IT staff runs around putting temporary fixes (band-aids) on issues instead of working to prevent the issues in the first place. The three most common methods of reactive management are:

- Use of monitoring software
- Performing backups
- Replicating data

Proactive management is putting solutions for disaster avoidance in place before there is a disaster. This minimizes the downtime associated with a disaster and may prevent many small issues from becoming major disasters.

The main ways to provide proactive management are:

- Building fault tolerant servers
- Following the vendors' (software or hardware) best practices
- Traditional clustering solutions
- True high availability solutions

In the following paragraphs, we'll explore each of these options. When comparing the various methods, keep in mind these questions:

- What type of failure does this solution protect against?
 - Configuration Problems
 - Data Loss or Corruption
 - Application Failure
 - Network Outages
 - Poor Performance
 - Server Failure
 - Site Outages (Disasters)
 - Does this solution keep my users connected during the failure?

Use of Monitoring Software

Monitoring software watches your systems and alerts you when there is a problem. However, it does not do anything to fix the problem. You must still take action to get the failed system back online. Monitoring software is only as good as the person configuring the alerts. This makes this software very prone to "poor configuration" disasters.

A poorly configured monitor will either post too much information or not enough. Too much information can lead to false negative alerts. When you start receiving a lot of "bogus" messages you may start ignoring them which kills the usefulness of the monitoring software. If you don't get enough information, you may not know about a problem until it is too late. If you don't mind spending countless hours configuring your monitoring software then you may someday tweak it to the point of being useful. Just remember it is completely reactive. It does not protect you against failure or keep your users connected during the failure. It just reports what is happening.

Performing Backups

Backups work by providing a redundant copy of your data. Typically these copies are stored on some type of tape device. There are many variations of backup hardware and software available. The type of hardware determines the amount of data and the speed at which it can be backed up. Each software package has its own unique advantages such as detailed reporting, ease of use, functionality, etc.

Depending on the amount of data, backups can take a long time to restore and your systems are down during the entire process. Backups only protect against data loss. If your application fails, you must either fix the application or reinstall it and then restore the data afterwards. The data contained in your backup is usually stale by the time you restore it. Unless your server fails immediately after you back it up you will lose some data. Backups should be the last option for recovery as your users are disconnected from the server during the restore process – resulting in extended periods of lost productivity and revenue.

Replicating Data

Like backups, replication software only protects against data loss. Replication software is much more efficient than backups as it allows for scheduled snapshot style replication, as well as asynchronous real time replication. This keeps an exact up-to-date copy of your data on another machine. Some replication software will even redirect requests to the backup copy when a hardware failure is detected on the production server. However, very few will monitor for network, application or performance problems, and all typically require an end-user to restart their applications in order to reconnect to the failover server.

Most replication products only replicate in one direction. Meaning once you fix the problem with your production server you can't just replicate the data back. You have to backup the data on your replication server, restore it to your now working production server, and then setup replication again. This makes it very time consuming to get your production environment back to normal after a failure. Also, it requires that your users be disconnected from the production server while you are putting things back to normal.

Building Fault Tolerant Servers

It's a good business continuity practice to build as much fault tolerance as possible into your servers. Configure them with redundant power supplies, fans, NICs, CPUs, etc. Install RAID controllers. Use at least RAID 1 or RAID 5 for your critical data. Just remember, this alone is not enough to protect you against disasters. This should be your first line of defense, but not your only line of defense. Building fault tolerant servers can help with server failure and poor performance due to hardware issues, but it does not protect against any other type of failure. Once there is a failure, you must rely on some other method to get the server back online.

Following the Vendor's Best Practices

Try to follow the vendor's recommendations on how to deploy your hardware, network and software. By following these best practices you ensure that your system is operating as efficiently as possible. The difficulty in following best practices is that it takes either a lot of skill or a

lot of research time to do it correctly. You could bring consultants in to configure your systems for you, but it is going to cost you more and that knowledge is lost when the consultants leave. Following best practices can help with failures due to configuration problems and with performance issues, but once there is a failure you must use another method to recover your server.

Traditional Clustering Solutions

Clustering works by having multiple machines (called nodes) serving the same data. The data is either stored on a shared storage device between the nodes or is replicated between the nodes' individual local storage. One of the nodes (called the active node) services users while the other node (called the passive node) waits in case of failure. If the active node has a problem then the passive node takes over. This process is called failover. Usually this only takes a few minutes and then your users are automatically reconnected back to their applications. However clustering solutions are complex and costly to implement making this solution out of reach for most businesses.

True High Availability Solutions

A high availability solution provides constant connectivity to critical applications regardless of the reason for the failure. It protects against foreseen and unforeseen problems. It should not require restarting applications and should provide maximum uptime with practically no downtime. A good high availability solution should provide the benefits of monitoring software, backups, replication software, and clustering all in one package.

When comparing high availability solutions you need to ask yourself the following questions:

- What hardware is supported with this solution?
- Is this solution affordable?
- Does this solution support all of my applications?
- Is this solution easy to implement?
- Does this solution provide disaster avoidance and disaster recovery?

Neverfail High Availability Solutions

Neverfail provides affordable, true high availability and disaster recovery solutions tailored for the Windows-based technology platform. Neverfail's solutions keep all users seamlessly connected to key applications, regardless of the source of failure. This includes disruptions due to poor configurations, network outages, operating system issues, application failures, hardware problems, or any of the other disasters mentioned in this whitepaper.

Unlike a traditional cluster that waits for a failure before it reacts, Neverfail is proactive in looking for problems. Neverfail SCOPE (Server Check Optimization Performance Evaluation) software automatically analyzes your server for information about services, third party applications, software and system configuration, disk space and performance measurements. This level of detail lets you see how reliably your system is functioning and can alert you to issues before they become problems.

Neverfail synchronizes the data between local storage on each node in real time. This provides the high availability of clustering with the hot backup functionality of data replication. It also removes shared storage as a single point of failure between the nodes and tremendously reduces the expense of deploying a high availability solution. Local storage is relatively cheap when compared to the cost of implementing and supporting a SAN.

Neverfail provides high availability over LANs and WANs. You can have your nodes spread out geographically to protect against complete data center disasters or network outages. In situations where bandwidth or network utilization needs to be reduced, the Neverfail Low Bandwidth Module (LBM) is available to reduce the amount of data transmitted between nodes.

Using Neverfail doesn't add much to the hardware requirements for your high availability solution. You need at least 2 network cards per machine, 2 GB of free disk space, and ideally up to 1 GB of RAM. Neverfail doesn't require each node to be identical. The only requirement is that the passive node has the same version of the operating system installed and at least the same amount of memory and disk space as the active node. This can save you money by allowing you to use an older or less powerful server for the passive node.

Installing Neverfail is very straight forward. In a nutshell, you load the operating system onto the passive node. Then configure and connect the network cards and install the Neverfail software. Once you install the Neverfail software, it sets up the passive node for you. In simple terms, Neverfail works by cloning your active server to your passive server. It automatically installs all of the applications, renames the server, and configures the networking settings. The Neverfail software then hides the passive node from the network.

Neverfail supports SharePoint, Exchange, SQL Server, IIS, BlackBerry, File Server, Oracle database and Lotus Domino applications right out of the box. It also protects auxiliary applications such as anti-spam, anti-virus, backup and fax. Neverfail provides an API (application programming interface) to provide protection for virtually any application. Numerous products are supported including IBM Websphere MQ, McAfee GroupShield for MS Exchange, Sophos MailMonitor, Sun Accounts, and Symantec Mail Security for Microsoft Exchange.

In addition to disaster avoidance, Neverfail provides disaster recovery as well. It protects against data corruption by rapidly restoring application data and configuration settings to a previous point in time with the Neverfail Data Rollback Module.

The Neverfail Data Rollback Module utilizes Microsoft's Volume Shadow Copy Service (VSS) technology. It takes shadow copies of the passive node and stores up to 512 snapshots at any given time. When a rollback is needed Neverfail restores the required application data and configuration settings from shadow copies, rather than restoring an entire volume as with traditional backups. You can customize your own snapshot schedules and decide how long each snapshot should be saved.

Putting It All Together

Companies these days are more dependent on computers than ever before. Think of how your company would be affected if your mission critical server(s) went down. How much in profits would be lost due to reduced productivity? How would downtime affect your company's reputation?

There are many types of disasters, both large and small, common and not so common. Your goal as an administrator should be to minimize the risks associated with these disasters. This can be accomplished through a combination of reactive and proactive methods. The reactive methods include monitoring your servers and backing up your data with backup or replication software. The proactive methods are building fault tolerant servers according to the vendor's best practice, clustering, and high availability solutions.

Build your servers with fault tolerance and best practices in mind, but don't let this be where your disaster avoidance/recovery planning stops. You should always backup your servers in case of a total disaster. Just don't solely rely on backups. Instead implement a high availability solution to provide constant connectivity to your users.

A true high availability solution should not require restarting applications and should provide maximum uptime with practically no downtime. It should monitor your server to prevent disasters when possible allow you to recover as quickly as possible when there is a failure. It should provide the ability to roll back data for the instances when the failure is due to data corruption, rather than application or hardware problems.

When putting your disaster recovery plan together use this white paper as a checklist to compare solutions. Decide the types of disaster for which your company is vulnerable. Compare what you stand to lose in actual dollars to what a high availability solution costs. Depending on your industry, avoiding a few hours of downtime can completely pay for the costs of avoiding disasters in the first place.