"Dimitrie Cantemir" Christian University
Knowledge Horizons - Economics
Volume 6, No. 3, pp. 45–50
P-ISSN: 2069-0932, E-ISSN: 2066-1061

www.orizonturi.ucdc.ro

IMPROVING INFORMATION SECURITY BY IMPLEMENTING FAULT TOLERANCE CONCEPTS

Aurel ȘERB

Faculty of Finance, Banking and Accounting, Bucharest, "Dimitrie Cantemir" Christian University, E-mail: aserb@ucdc.ro

Abstract *Security issues are complex, and the risks of cyber crime are often difficult to stipulate, even for experts. The issues presented in this article try to be formed in a contribution to the consolidation of problems in the field of computer arhitecture cyber security. Fault tolerance is the best guarantee that high-confidence systems will not succumb to physical, design, or human-machine interaction faults. A fault tolerant system is one that can continue to operate reliably by producing acceptable outputs in spite of occasional occurrences of component failures. A fault tolerant cluster is a cluster with a set of independent nodes, connected over a network, and always with external storage devices connected to the nodes on a common input/output bus. The cluster software is a layer that runs on top of local operating systems running on each computer. Clients are connected over the networks to a server application that is executing on the nodes. The nodes of a cluster are connected in a loosely coupled manner, each maintaining its own separate processors, memory, and operating system. Special communications protocols and system processes bind these nodes together and allow them to cooperate to provide outstanding levels of availability and flexibility for supporting mission critical applications.*

Key words:
Cyber Security, Fault tolerance, Fault Tolerant Cluster, Single Point of Failure, High Level Architecture

JEL Codes:
C63

1. Introduction

Advances in information technology have created major efficiencies in the design of systems. Hardware is constantly improving and more sophisticated and powerful software packages are available commercially. And the ability to network both locally and over worldwide systems can be fully utilized for the management of these systems.

The natural growth in scope and importance of the computing environment has been accelerated by recent trends toward globalization and mergers. The increasing reliance on networked applications and information means that all parts of an organization can be seriously affected by physical, design, or human-machine interaction faults, by viruses and malicious acts or by an isolated local disaster such as an earthquake, flood, hurricane, fire, or theft.

Many users are still not aware of the possible vulnerabilities, threats and risks they meet by using computers, software and software application and communication networks or solutions that already exist to make them face. Security issues are complex, and the risks are often difficult to stipulate, even for experts. Lack of information is one of the imperfections of the market, on which security policies should be to have in mind. There is a risk that some users alarmed of so many reports and threats to security, to avoid simply the use some of the information systems. Others, which are either not scripturally informed, or underestimate the risk, may be too negligent. Paradoxically, there is an impressive quantity of information in the field of computers, networks and security of the information available on the Internet and magazines about computers covers the subject quite well.

2. The Definition of Informatics Security[1]

ISO/IEC 27002 is the international regulation used when reference is made to issues that have connection with cyberspace security (cyber security). This standard analyzes three different concepts as a complex which defines cyberspace security, and thus defines different computer security, information security and assurance computing, all of which are components of information security:

- *Computer Security* is a branch of computer science known as information security applied to computers and networks. The main objective of computer security is the protection of information and property against theft, corruption, or natural disaster, allowing at the same time, the information and property to remain accessible and productive for their users. Term security of computer systems

refers to the collective processes and mechanisms by which information and confidential services, or value, to be protected from publication, tampering or destruction by unauthorized, unauthorized or unplanned events. Computer Security focuses on the availability and correct operation of a computer system, without regard to the information stored or processed by the computer.

- *Information security* means protecting information and information systems from unauthorized access, use, disclosure, penetration, modification, or unauthorized destruction.
- *Information assurance* refers to risk management related information, specifically to protect and defend information and information systems by ensuring confidentiality, integrity, authentication, availability and non-repudiation. These objectives are pursued whether the storing, processing, and transfer of information, or inconvenience caused by bad faith, or due to accidents.

3. Fault Tolerance[2]

Only ideal systems would be perfectly reliable and never fail. This, of course, is impossible to be achieved in practice, because the systems fail for many reasons. Fault tolerance is the best guarantee that high-confidence systems will not succumb to physical, design, isolated local disaster, or human-machine interaction faults.

A fault tolerant system is one that can continue to operate reliably by producing acceptable outputs in spite of occasional occurrences of component failures.

The basic principle of fault-tolerant design is the use of redundancy, and there are three basic techniques to achieve fault tolerance: spatial (redundant hardware), informational (redundant data structures), and temporal (redundant computation).

The classical hardware and software fault tolerant techniques are modular redundancy, N-version programming, error-control coding, checkpoints, rollbacks, and recovery blocks.

Modern systems are partitioned at several levels based on functions provided by specific subsystems. A fault-tolerant system displays similar functional partitioning, but in addition it contains redundant components and recovery mechanisms which may be employed in different ways at different levels. It is reasonable to view a fault-tolerant system as a nested set of subsystems each of which may display varying levels of fault tolerance. Recovery from a fault within a redundant partition may be affected within the domain itself, or may require action by higher levels within the system.

Fault tolerant architectures package these redundant partitions into replaceable units. A replaceable unit is a unit of failure, replacement and growth - that is, a unit that fails independently of other units, which can be removed without affecting other units, and can be added to a system to augment its performance, capacity, or availability.

The desired result of system partitioning and subsystem design is an integrated set of local, intermediate, and global fault tolerance functions that serve as a protective infrastructure to ensure the timely and correct delivery of system services.

4. Clusters and Fault Tolerant Clusters

A distributed system is a collection of computers (called nodes) that communicate with each other through a communication medium. Under the control of systems software, the nodes can co-operatively carry out a task. An open system allows system integration, so the customers can choose various hardware and software components from different vendors and integrate them to create a custom configuration suiting their needs and cost requirements.

A cluster is a set of loosely coupled, independent computer systems, connected over a network that behaves as a single system. The cluster software is a layer that runs on top of local operating systems running on each computer. Client applications interact with a cluster as if it is a single high-performance, highly reliable server. System managers view a cluster much as they see a single server. Most applications will run on a cluster without any modification at all. And only standard-based hardware components such as SCSI disks and Ethernet LANs are used to create a cluster.

Clustering can take many forms. A cluster may be nothing more than a set of standard personal computers interconnected by Ethernet. At the other end of the spectrum, the hardware structure may consist of high-performance symmetric multiprocessor systems connected via a high-performance communications and I/O bus. In both cases, processing power can be increased in small incremental steps by adding another commodity system.

If one system in a cluster fails, its workload can be automatically dispersed among the remaining systems. This transfer is frequently transparent to the client.

A fault tolerant cluster is a cluster with a set of independent nodes, connected over a network, and always with external storage devices connected to the nodes on a common input/output bus. Clients are connected over the networks to a server application that is executing on the nodes. The nodes of a cluster are connected in a loosely coupled manner, each maintaining its own separate processors, memory, and

operating system. Special communications protocols and system processes bind these nodes together and allow them to cooperate to provide outstanding levels of availability and flexibility for supporting mission-critical applications. Fault tolerant clusters maintain strict compliance to the principles of open systems. There are no proprietary application programming interfaces that force vendor lock-in and require substantial development investment. Most applications will run on a fault tolerant cluster without any modification at all.

The top-level software of a fault tolerant cluster can be designed to maximize the flexibility of configurations within a local cluster. Clusters may be formed with a different number of nodes. This flexibility in system selection and cluster configuration protects customer investments in installed systems and allows the processing power of each node to be matched with the specific requirements of each application service.

If the failure of any component in a cluster results in the unavailability of service to the end user, this component is called a single point of failure for the cluster. One of the most important problems in implementing fault tolerant system is the identification of single points of failure and elimination of these single points of failure by using replaceable units.

The elimination of a single point of failure, by using replaceable units, always has a cost associated with it. Usually, what can be done is only to attempt to make a service highly available if the cost of losing the service is greater than the cost of protecting it.

The possible single points of failure that a cluster could have are:

– *Nodes in the cluster,*

– *Disks used to store application or data, adapters, controllers and cables used to connect the nodes to the disks,*

– *The network backbones over which the user are accessing the cluster nodes and network adapters attached to each node,*

– *Power sources,*

– *Applications.*

There are two typical clustering topologies that provide high availability that are looked at in this paper: passive backup server, and active/active server. Three versions of the active/active server are looked at in more detail: “duplicate everything,” “share nothing,” and “share everything.”

Each of these clustering topologies has a “heartbeat” mechanism integrated into each server that serves to confirm server viability. The sending or primary server sends periodic messages to the alternate or backup server. If the messages should stop, the viable server assumes that the other server has failed and put itself into operation in place of the failed server.

4.1 Pasive Backup Server

A common arrangement is for one server to act as the primary server, with a secondary server available for use should a failure occur in the primary server. With the passive backup server approach (see Figure 1), a secondary server is not used for any other processing but it simply stands by to take over in the event of a failure of the primary server.

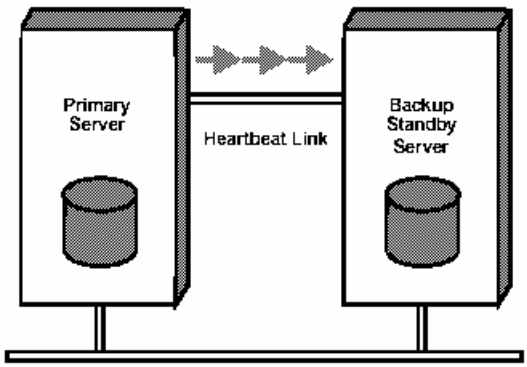

Figure 1. Clustering Using a Passive Standby Server

4.2 Active/Active Servers

A related but far more cost-effective approach is to enable the additional server to perform useful processing on other applications but still be able to take over for the initial server in the event it should fail. The key benefit of this approach, which is called the active/active server approach, is that customers can have server redundancy while retaining use of each server, instead of limiting it strictly to a backup function. This method can reduce the overhead costs of operating a clustered system. Three variations of the active/active server approach are discussed further.

4.3 Duplicate Everything

One type of solution is to duplicate everything. That is, have entirely redundant servers with their own disks. With this approach (see Figure 2), data must be constantly copied to the disks of the secondary system to ensure that should a failure occur, the secondary system has access to current data. Although this approach does deliver high availability, it can result in substantial overhead on the servers and the LAN, which may impact performance. A second drawback is that there will be some delay in transferring information and user connections from one server to the other.

Redundant servers provide several benefits; since the data is completely replicated, the client's applications can access either server, allowing for better load balancing. Additionally, the nodes can be geographically dispersed; the node connection can be with a WAN and the nodes can be physically far away from each other. This topology is frequently used as part of a disaster recovery plan, providing protection against major catastrophic events like earthquakes or floods.

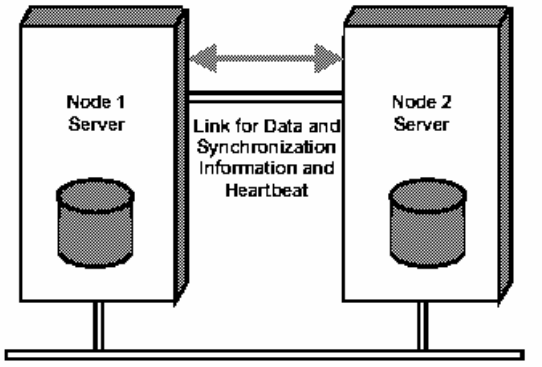

Figure 2. Clustering Using the "Duplicate Everything" Approach

4.4 Share Nothing

The second variation of the active secondary server approach is to share nothing. That is, while the two servers are physically cabled to the same disks, only one of the servers owns the disks at any time. With this approach, each server owns its own set of disks, and during normal operation only the owner server may access its disks (see Figure 3). However, in event of a server failure, an alternate server (in the cluster) can assume ownership of the failed server's disks and access them.

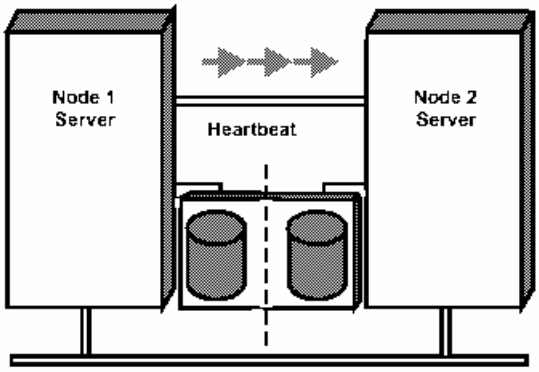

Figure 3. Clustering Using the "Share Nothing" Approach

This approach eliminates the need to constantly copy data over the LAN to a secondary system, significantly reducing server and LAN overhead. With this approach, the disks could become a single point of failure such that a disk failure could result in an extended period of downtime. Thus, installations using this architecture typically rely on redundant mirrored disks (RAID technology) in the disk subsystem to ensure availability of applications and data in the event of either server or disk failure.

4.5 Share Everything

Finally, it is also possible to share everything, that is, has multiple servers share the same disks at the same time (see Figure 4). With this approach, during normal operations, all servers connected to the disks can share access to them at the same time. This approach requires development of sophisticated lock manager software to ensure that only one server at a time can access the data. The "share everything" approach again typically relies on mirrored (RAID technology) disks.

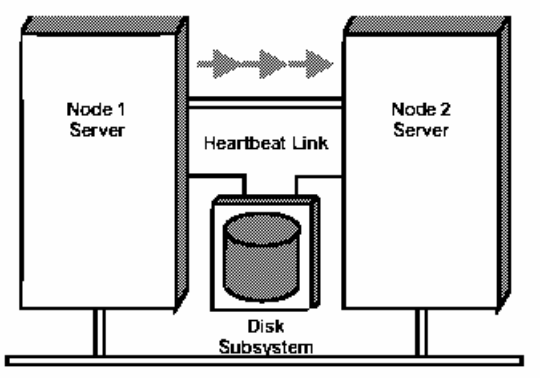

Figure 4. Clustering Using the "Share Everything" Approach

5. Conclusions

One of the most important purposes of clustering is to increase system availability. Generally, a high-availability configuration ensures that if a server or application should unexpectedly fail, another server in the cluster can both continue its own work and readily assume the role of the failed server. The goal is to minimize server and application downtime to end-users.

"Load balancing" is a technique that allows the performance of a server-based program to be scaled by distributing its client requests across multiple servers within the cluster. Application packages are the usually way for providing the powerful and flexible mechanism for balancing workload within the cluster after a node failure. Individual application packages within a single node can be moved to different nodes, distributing the workload of one node across the surviving nodes of the cluster. If a node fails, each of the packages running on that node can be moved to a different node. This distributes the workload of the failed node among all of the remaining nodes of the cluster, minimizing the performance impact on the other applications within the cluster. After the first node is made healthy and reintroduced into the cluster, fault tolerant software automatically assigns this first node as the new standby node With the automatic failback functionality configured, applications automatically move back to the primary node once it is made healthy and reintroduced to the cluster. This ability to move application packages also allow the workload of a cluster to be balanced even when there is no failure.

Another benefit derived from clustering is system scalability Clustered systems can be expanded when the overall load exceeds the capabilities of the original systems. In this case, the additional servers can be plugged into the cluster and will contribute to the overall application's performance in the cluster.

Server clusters are complex, and complex technologies can introduce many more opportunities for human error. For these reasons, system operations should be monitored and administered by a network resource management system, which provides tools to simplify the management process. Effective management of server clusters requires an integration of view of a cluster as a single system and as separate servers.

Clustering for high availability addresses both planned and unplanned downtime. For instance, when a system manager needs to backup or service a system, the activity can be planned for a time that will be less disruptive for the end users. Unplanned downtime is by definition all the downtime that cannot be scheduled, such as, a system hang or a hardware failure. Clustering software will detect the failure and move the application to an alternative server with minimal interruption to the user.

References

[1] Șerb A. (2010). *Securitate informatică* – Editura Pro Universitaria, București.

[2] Șerb A., Baron C., Isăilă N., Ionescu C., Defta C. L. (2013). *Securitatea informatică în societatea informațională* – Editura Pro Universitaria, București.

[3] Șerb A., Patriciu, V. V. (2000). *Using of Fault Tolerant Distributed Clusters in the Field of Command and Control Systems* – NATO's Research & Technology Organization PfP Symposium "New Information Processing Techniques for Military Systems", Istanbul, the Turkey, 09-11 October.

[4] Șerb A. (2010). *Sisteme de calcul tolerante la defectări,*Editura Academiei Tehnice Militare, București.

[5] Patterson D.A., Hennessy J.L. (1998). *"Computer Organization & Design. The Hardware/Software Interface"* - Morgan Kaufmann Publishers, Inc., San Francisco, California, U.S.A.

[6] Weygant P. (1996). *"Clusters for High Availability. A Primer of HP-UX Solutions"* - Prentice Hall Pt., Upper Saddle River, New Jersey, U.S.A.

[7]. Șerb A. (1999). *"Fault Tolerance in Systems Used for Computer Assisted Exercises"*, NATO's Research & Technology Organization PfP Symposium on Computer Assisted Exercises for Peace Support Operations, The Hague, the Netherlands, 28-30 September.

[1] Şerb A. (2010). *Securitate informatică* – Editura Pro Universitaria, București, 2010.

[2] Şerb A., Patriciu, V. V. (2000). *Using of Fault Tolerant Distributed Clusters in the Field of Command and Control Systems* – NATO's Research & Technology Organization PfP Symposium "New Information Processing Techniques for Military Systems", Istanbul, the Turkey, 09-11 October.