클라우드 기반의 재해 복구 시스템

매니쉬*, 최승준*, 이봉재*, 박종서*

요약

재해는 시스템의 생존주기 동안 일어날 수 있는 원하지 않는 일이다. 시스템을 구축할 때, 재해를 예방하고 최소화하며 가능한 많은 차선책을 확보하거나, 재해가 발생하였을 때 복구 되는 시간을 최소화 하는 것이 필요하다. 재해 복구에 대한 기존의 접근 방법이 있지만 손실된 전체 데이터를 완전히 복구하지 못하고 정상적인 상태로 복구하는 시간도 많이 걸리게 된다. 여기서는 기존의 정보통신기술(ICT: Information Communication Technology) 기반의 시스템에서 사용된 재해 복구 방법을 분석하여 문제점을 찾고, 이러한 문제점에 대한 해결책을 제시하고자 한다. 본 논문에서는 클라우드 컴퓨팅기반의 ICT 기반 시스템에서 재해 복구방안에 대해 논하고자 한다.

The Cloud-Based Approach for Disaster Recovery System

Manish Pokharel*, Seung-Jun Choi*, Bong-Jae Lee*, Jong-Sou Park*

ABSTRACT

A disaster is unwanted event that may occur at any time during the life time of a system. Once a system is built, it needs to be able to prevent and minimize the event and have several alternatives as many as possible or work around such events as much as possible, or at least it should be recovered quickly when the event occurs. There are few existing approaches in recovering from such unwanted events but they do not recover the entire lost data and also take more time to be back to the operating state. We analyze the existing disaster recovery approach used in Information Communication Technology based system and find the loopholes in it, then try to get the solution to fix these loopholes. In this paper we address disaster recovery plan for an Information Communication Technology based system based upon the cloud computing.

Key Words : System, Cloud Computing, Disaster, Disaster Recovery, Availability, Survivability

* 한국항공대학교 컴퓨터공학과(✉manishpokharel@gmail.com)
· 제1저자(First Author) : 매니쉬(Manish Pokharel) · 교신저자(Correspondent Author) : 매니쉬(Manish Pokharel)
· 접수일(2011년 3월 7일), 수정일(1차 : 2011년 5월 2일), 게재확정일(2011년 5월 6일)

Ⅰ. Introduction

A disaster refers to any unexpected and unwanted events that render threats to life and property or causes the disruption of the normal functioning of a system. Such events could be the result of natural disasters, technical faults or human made errors/disasters. The earthquake, flood, volcano, Tsunami are natural disasters. Mechanical failures of equipments, software bugs are the causes of technical faults, where as terrorist attack like 9/11, 6/6 are human made disaster. Such events are not generally expected or do not have calendars but we have to be careful about our recovery plan for it. The disaster recovery and cloud computing are described in Introduction. We put the current work done in this area in section 2 as literature review. The existing problems of disaster recovery are given in section 3 and propose the solution as a cloud based approach in section 4. We refer our previous work done and make the continue works on it. [1].

We analyzes the existing system with Markov model and try to compare it with the proposed system. We consider the availability, survivability and downtime as comparing attributes between existing system and proposed system. These attributes make the system dependable and trustworthy. We try to emphasize on disaster recovery system and provide the best solution on it with the comparison between with cloud and without cloud in section 5 and finally conclude the paper in Section 6.

Ⅱ. Literature Review

Trust is one of the prime factors in convincing the stakeholder. They trust the system if it can provide services all the time, survive during fragile condition, and does not go into down state for a long time. Many efforts and researches have been conducted to convince the stakeholder. The status quo of existing system is not very much convincing. The existing system claims the reliability, security, and disaster recovery in some extent but the level of such features are not convincing. The reliability is achieved with just extra redundancy system. The security is achieved up to some level through the traditional security approaches. Disaster Recovery is carried out by providing physical backup system far away from the main operating system's location, which takes significant time to restart the system. A concept of GIS and safe communication channels has been used to design disaster recovery system for ICT based system especially in e-government system. The approach is very primitive and not applicable to the range of e-government system [2]. Distributed system and meta data handling concept was proposed for disaster recovery in which there are three main components like data center, supervisory mode and client nodes. In this approach the incoming data files are encrypted, fragmented, duplicated and finally distributed [3]. Disaster recovery can be addressed with some contingency planning using virtualization technology. Virtualization technology offers the advantages like portability, hardware independence, survivability, availability and also reliability. These properties are essential for any system in its life time and especially during disaster. Stephen C. Gay has put the possibilities of using virtualization technology in managing disaster recovery[4]. Even though none of

them is satisfactory enough to be convinced but virtualization technology provides the possibility of working more on these areas. In this paper, we propose the disaster recovery system and try to address the existing problems on disaster recovery through cloud computing approach.

III . Problems in Traditional Disaster Recovery Plan (TDRP)EA

Numbers of different Disaster Recovery Plans (DRP) are being used since long time, among which some of them are effective and some are not. We classify the history of disaster recovery into three phases; Disaster Recovery in Past, Disaster Recovery in Present and Disaster Recovery in Future. We keep the past and present form of disaster recovery under the traditional DRP category and the future of disaster recovery is cloud-based recovery. Here, we first mention the traditional DRP in very brief and then propose new DRP based upon the cloud computing approach.

3.1 Traditional DRP

In most of the traditional DRP's system, multiple replication of a same system is made and placed into different geographic location as a copy of primary system. If disaster occurs in primary location then the secondary location will provide the information, but rarely services. Mostly in traditional approach, this is done in two ways. One is online backup i.e. if the system is down then another same type of system will be up and provide the information, which is also known as Disk Mirroring. Another approach is offline backup in which the system of another location does not start immediately like online but the data are brought back to the compromised system and wait till the system gets ready to take data and not necessarily it provides both information and services.

3.2 Analysis: Without Cloud

In this section, we try to do analyze the disaster recovery method without using cloud computing. We follow CTMC (Continuous Time Markov Chain) for analyzing the system. Here, we consider the traditional disaster recovery approach in which there are many disks, tapes used to back up the data and are kept in far distance from the main system. Here, we assume one week is required to get the backup data and start the system.

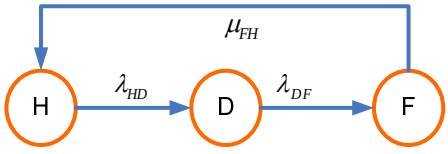

그림 1. 클라우드 컴퓨팅을 제외한 상태흐름도
Fig. 1. State Diagram without using Cloud Computing

In above Figure 1, there are three states i.e. Healthy, Defect and Failure. A system provides service if it is in healthy state and in defect state but no service in failure state. There is no mechanism to recover when it is in defect state except getting backup data from another location. The time it takes to move from defect state to failure state is assumed to be very high i.e. 5 hours so the failure rate is 0.2 hours. We assume the system works for 5000 hours i.e. MTTF is 5000 hours. We take same working hours for our next analysis with cloud computing in

section 4. We consider the same system but the different is only the way we handle disaster. This is the case of handling without cloud computing and another is with cloud computing.

Let's start with analysis each state of above Figure 1 with steady state probabilities of every state with balanced equation.

State"Healthy"

$$\pi_H \lambda_{HD} = \pi_F \mu_{FH} \quad \text{(A)}$$

State"Defect"

$$\pi_H \lambda_{HD} = \pi_D \lambda_{DF}$$

$$\pi_D = \frac{\lambda_{HD}}{\lambda_{DF}} \pi_H \quad (1)$$

State"Failure"

$$\pi_F \mu_{FH} = \pi_D \lambda_{DF}$$

$$\pi_F = \frac{\lambda_{HD}}{\mu_{FH}} \pi_H \quad (2)$$

Balance Equation:

$$\pi_H + \pi_D + \pi_F = 1 \quad (3)$$

Now,

$$\pi_H = \left[1 + \frac{\lambda_{HD}}{\lambda_{DF}} + \frac{\lambda_{HD}}{\mu_{FH}} \right]^{-1} \quad (4)$$

As per the above state diagram and the nature of the existing system, we use following operating parameters given in Table 1.

표 1. 작동 매개변수
Table 1. Operating Parameters

Parameters	Meaning	Values
λ_{HD}	Defected Rate	0.0002 hours
λ_{DF}	Failure Rate	0.2 hours
μ_{FH}	Repair Rate of Failure	0.00595 hours

3.3 Calculation

We use the parameters' values from Table 1 and put these values in equation (1), (2) and (4), we get the steady state probability of every state. Steady state probability is the probability of every state to be in its own state. It is the inertia of the state given in Table 2. We use simulation tool for this purpose. We have verified these values with a SHARPE (Symbolic Hierarchical Automated Reliability and Performance Evaluator), is a well known package in the field of reliability, availability and per formability. The SHARPE is one of the appropriated simulation tool for evaluating non functional properties of a system. With these values we calculate the availability, survivability and downtime of this system in this section below.

표 2. 모든 상태의 확률
Table 2. Probability of every state

π_H	0.9665
π_D	0.0009665
π_F	0.03248

3.3.1 Availability

It is the probability of existing disaster recovery system to provide the service. Availability of the system is 1- Unavailability

The service will be unavailable only if it is completely fail i.e. if it is in failure state. So, the availability can be calculated as:

Availability = 1 - π_F

Availability = 1 - 0.03248

Availability = 0.96752 [This means service is available 96.7% of the service time.] (5)

3.3.2 Survivability

Here we find out the probability of system to be survived in spite of disaster. We have the equation for the survivability given below.

Survivability = Availability - π_D

Survivability = 0.96752 - 0.0009665

Survivability = 0.96655 [This means survivability is 96.65% of the service time.] (6)

3.3.3 Downtime

This is the time when the entire system goes down. The calculation is done with following equation. We consider the time interval as one year.

Downtime = Probability of system to be in failure state * Time Interval (L)

We assume L as 1 year i.e. (12 x 30 x 24) hours, so total down time in a year would be:

Downtime = 0.03248 x (12 x 30 x 24)

Downtime = 280.62 hours [This means downtime of the proposed system is 280 hours in a year.]

3.4 Demerits of Traditional DRP

The main demerits of traditional disaster recovery system is high cost, more resource utilization, complex infrastructure, security threat and low availability and survivability. Here, we propose cloud-based Disaster Recovery Plan in the following section to overcome the problems of traditional approach.

IV . Using Cloud Computing

Here, we use cloud computing for disaster recovery. All required features are provided as a service.

4.1 Developing DRP: Proposed Approach

Based upon the nature of the system, there are different approaches in developing DRP in cloud computing.

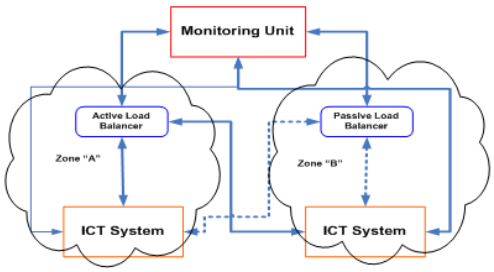

그림 2. 클라우드 컴퓨팅을 적용한 실패 복구

Fig. 2 Disaster Recovery with Cloud Computing

We propose above approach given in Figure 2 for disaster recovery plan for our ICT system. In above Figure 2, there are two zones i.e. Zone "A" and Zone "B". Each zone is replication of another i.e. both zone contains the same type of system. If one zone gets down due to disaster then another will be up with the data and services both. The service users will not be deprived by the services if disaster happens. Disaster like flooding, hurricane, make system underwater for a long time and takes much time to renovate it. Our proposed disaster recovery approaches take care of such incidents.

In Figure 2, both zones have systems. The zone "A" has Active Load Balancer to balance the work load of the system. Resources are assigned as per the degree of works which is nothing but scalability features of cloud computing. Zone "B" has a Passive

Load Balancer whose functions are same as Active Load but it resides in passive mode. Zone "B" gets active if Zone "A" does not work. Monitoring Unit monitors both zones and performs actions accordingly especially switching between two zones.

Analysis:

For analysis, we draw the state diagram of our proposed disaster recovery mechanism. The diagram is given below.

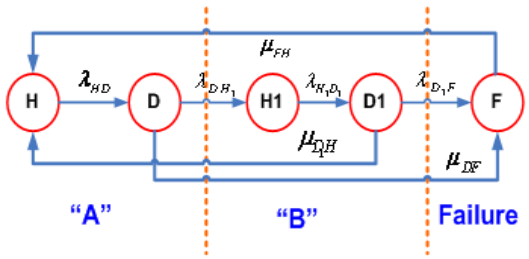

그림 3. 클라우드 컴퓨팅을 적용한 상태 흐름도
Fig. 3 State diagram with Cloud Computing

There are five states in Fig 3 with two zones i.e. A and B. Here, we use Continuous Time Markov Chain (CTMC) for analyzing. Service is provided if it is in state H. If there are some problems then it moves to defect state with rate λ_{HD}. This is the base longevity interval. This transition from healthy state to defect is being monitored by Monitoring Unit as given in Figure 2. The Monitoring Unit is smart enough to make a move from existing cloud to healthy state of another cloud i.e. B with switching rate λ_{DH_1} but it also has the chance for moving to failure state with μ_{DF}. We try to consider almost all possibilities of system going down. If B also gets defected being in state D1, we assume by this time A will be ready to provide the service after some repaired then Monitoring Unit switches to the previous cloud A with switching rate μ_{D_1H} at the same time it also has some chances to move to the failure state.

Here, we use same parameters with same values as Section 3.2. We use balance equation and keep the values of each state in equation 7.

$$\pi_H + \pi_D + \pi_{H_1} + \pi_{D_1} + \pi_F = 1 \qquad (7)$$

We get,

$$\pi_H = \left[1 + \frac{\lambda_{HD}}{\lambda_{DH_1}} + \frac{\lambda_{DH_1}.\lambda_{HD}}{\lambda_{H_1D_1}(\lambda_{DH_1} + \mu_{DF})} + \frac{\lambda_{DH_1}.\lambda_{HD}}{(\lambda_{D_1F} + \mu_{D_1H}).(\lambda_{DH_1} + \mu_{DH})} + \left\{ \frac{\lambda_{DH_1}.\lambda_{HD}.\lambda_{D_1F}}{(\lambda_{D_1F} + \mu_{D_1H}).(\lambda_{DH_1} + \mu_{DH})} + \frac{\lambda_{HD}.\mu_{DF}}{(\lambda_{DH_1} + \mu_{DF})} \right\} \right]^{-1} \qquad (8)$$

This is the probability of our proposed system to be in healthy state i.e. capable of providing services.

V . Final Comparison

We compare the disaster approach with and without cloud computing with some desired properties of a system. We consider availability, survivability and downtime as desired properties for a system to function it properly. The Table 3 below shows the differences in obtained attributes for disaster recovery system without using cloud computing and with using cloud computing. We plot the graph according to the data in Table 3 in Figure 4 and 5.

표 3. 비교표
Table 3. Comparison Table

Attributes	Without Cloud	With Cloud
Availability	0.96752	0.9999
Survivability	0.96655	0.999664
Downtime	280.62	1. 67

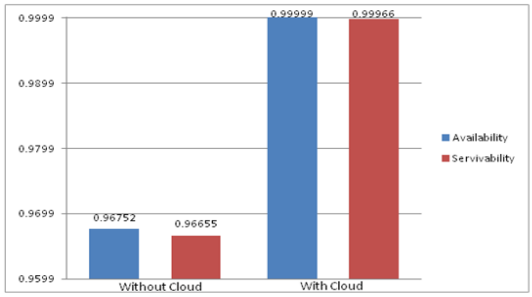

그림 4. 클라우드 컴퓨팅을 제외한 가능성/생존성

Fig. 4 Availability/Survivability without and with Cloud Computing

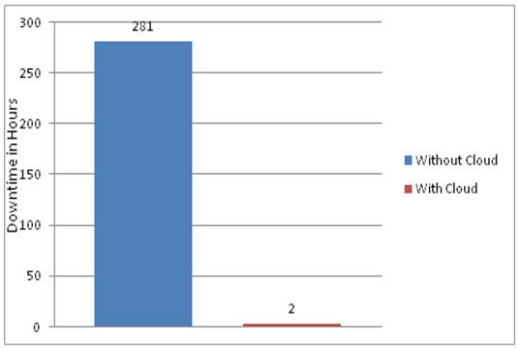

그림 5. 클라우드 컴퓨팅을 포함 여부와 중단시간

Fig. 5 Downtime without and with Cloud Computing

The Figure 4 is the obtained availability and survivability with and without cloud. We can see clearly the increase in availability of services and increases of survivability with cloud. Even the system is built for 5000 hours its availability is very less in without cloud computing approach. Our proposed cloud-based disaster recovery system obtains high availability and survivability. In Figure 5, it is the duration of system goes down after disaster. In traditional approach the system goes down for almost 281 hours where as in cloud-based disaster system the downtime is very low i.e. almost 2 hours. This figure shows the cloud-based disaster recovery system is more appropriate in system. We can even show the difference in the reliability but we have considered this work as our future research works.

VI . Conclusion

Disaster is an unpredictable event and thus a precise recovery plan is required to tackle such events. We have analyzed two types of disaster recovery approaches; one without cloud computing and another with cloud computing. In our proposed disaster recovery approach, we use Monitoring Unit that keeps on monitoring primary and secondary system site i.e. zone A and zone B. The Monitoring Unit ensures the possibility of high availability of services and can survive for a long time as compared to the traditional non cloud computing system because of its intelligent performance in observing, and switching between two zones. The cloud based disaster recovery system goes down only for 2 hours which is very low as compared to the downtime for traditional system. The traditional system needs almost 281 hours to be in working state.

참고문헌

[1] M. Pokharel, S. Lee, J.S.Park, "Disaster Recovery for System Architecture using Cloud Computing", 10th Annual International Symposium on Application and the Internet, SAINT , IEEE 2010

[2] K. Wang , G. Cai, Z. Li, L. Zhou ,"A Disaster Recovery System Model in an E-government System", International Conference on Parallel and Distributed Computing, Application and Technology,[PDCAT 05] IEEE 2005

[3] Y. UENO, N. MIYAHO, S. SUZUKI, "Disaster Recovery Mechanism Using Widely Distributed Networking and Secure Metadata Handling Technology", UPGRADE-CN' 09 ACM 2009

[4] S. Gay, "An Examination of Virtualization's Role in Contingency Planning", Information Security, Curriculum Development conference 07', ACM 2007

[5] R. Grossman."The Case for Cloud Computing", IEEE Computer Society

[6] L. Youseff, M. Butrico, and D. DaSilva, "Towards a Unified Ontology of Cloud Computing"Grid Computing Environments Workshop 2008, GCE08'

저자소개

Manish Pokharel

In 1995, Bachelor of Engineering in Computer Science, Karnataka University

In 2000, Master of Engineering in Software System, Birla Institue of Technology and Science

2007 ~ present Ph.D. student in Korea Aerospace University

※ Interest : E-government, Enterprise Architecture, Fault Tolerance and Cloud Computing

최승준(Seung Jun Choi)

Bachelor of Electronic Engineering, Korean Air Force Academy

Master of Electronic Engineering, Yonsei University

2010 ~ present Brigadier General of the Republic of Korea Air Force

※ Interest : Ubiquitous Sensor Network(MAC Protocol), C4I(Network and Application) System and Analysis and IEEE802.16e(Wibro, WiMAX)

이봉재(Bong Jae Lee)

In 1990, Bachelor of Communication Engineering, Korea Aerospace University

In 1993, Master of Information and Communication Engineering, Korea Aerospace University

2003 ~ present Ph.D. student in Korea Aerospace University

※ Interest : u-City, E-government, Information security.

박종서(Jong Sou Park)

In 1986, Master of Electrical and Computer Engineering, North Carolina State University

In 1994, Ph.D of Computer Engineering, Pennsylvania State University

1996 ~ present Professor in Korea Aerospace University

※ Interset : information security, embedded system and hardware design.