X Triennial International SAUM Conference

on

Systems, Automatic Control and Measurements

SAUM 2010

Proceedings

University of Niš
Faculty of Electronic Engineering
Faculty of Mechanical Engineering
Aleksandra Medvedeva 14, 18000 Niš
Serbia

November 10-12, 2010

Proceedings of the X Triennial International Conference on Systems, Automatic Control and Measurements
- SAUM 2010 -

Publishers:	Faculty of Electronic Engineering, Niš Faculty of Mechanical Engineering, Niš SAUM – Association of Serbia for Systems, Automatic Control and Measurements, Belgrade
Editors:	Prof. Dr. Vlastimir Nikolić Prof. Dr. Dragan Antić Dr. Darko Mitić
Technical Assistance:	Marko Milojković, M.Sc. Saša S. Nikolić Staniša Perić

ISBN 978-86-6125-020-0

CIP - Каталогизација у публикацији
Народна библиотека Србије, Београд

681.5(082)
62-52(082)
621.317(082)
53.08(082)
007:004(082)
007.52(082)

INTERNATIONAL SAUM Conference on Systems, Automatic Control and Measurements (10 ; 2010 ; Niš)
Proceedings / X Triennial International SAUM Conference on Systems, Automatic Control and Measurements, SAUM 2010, November 10-12, 2010, Niš ; [editors Vlastimir Nikolić, Dragan Antić, Darko Mitić]. - Niš : Faculty of Electronic Engineering : Faculty of Mechanical Engineering ; Belgrade : SAUM, 2010 (Niš : Unigraf). - 363 str. : ilustr. ; 29 cm

Tekst štampan dvostubačno. - Tiraž 150. - Bibliografija uz svaki rad.

ISBN 978-86-6125-020-0 (FEE)

a) Системи аутоматског управљања - Зборници b) Мерна техника - Зборници c) Мерни инструменти - Зборници d) Информациона технологија - Зборници e) Роботика - Зборници
COBISS.SR-ID 179422476

No. of copies: 150

Printed by: Unigraf, Niš

Proceedings of X Triennial International SAUM Conference on Systems, Automatic Control and Measurements

SAUM 2010

Dedicated to 50 years of Education in Electrical Engineering, Information Technologies and Mechanical Engineering in Niš

Niš, Serbia, November 10-12, 2010

Organization supported by

Ministry of Science and Technological Development
- Republic of Serbia

Organized by

SAUM – Association of Serbia for Systems, Automatic Control and Measurements

Co-organized by

University of Niš
Faculty of Electronic Engineering
Faculty of Mechanical Engineering

X Triennial International
SAUM Conference
on
Systems, Automatic Control and Measurements
SAUM 2010

Conference Chairman
Prof. Dragan Antić

International Program Committee

Prof. Vlastimir Nikolić, Serbia, president
Prof. Georges Bitsoris, Greece
Prof. Dragan Antić, Serbia
Prof. Pierre Borne, France
Prof. Zoran Bućevac, Serbia
Dr. Jovan Bosković, USA
Prof. Branko Kovačević, Serbia
Prof. Abdellah ElMoudni, France
Prof. Miroljub Grozdanović, Serbia
Prof. Toshio Fukuda, Japan
Prof. Lyubomir T. Gruyitch, Serbia
Prof. Zoran Gajić, USA
Prof. Milić Stojić, Serbia
Prof. Milorad Božić, Bosnia and Herzegovina
Prof. Srdan Stanković, Serbia
Dr. Sophie Hennequin, France
Prof. Bratislav Danković, Serbia
Dr. Alexander Ignatyev, Ukraina
Prof. Miodrag Arsić, Serbia
Prof. Zvi Retchkiman Konigsberg, Denmark
Prof. Novak Nedić, Serbia
Prof. Žarko Čučej, Slovenia
Prof. Vojislav Kalanović, USA
Prof. Dragiša Nikodijević, Serbia
Prof. Eberhard Kallenbach, Germany
Prof. Dušan Petrovački, Serbia
Dr. Annemarie Kőkösy, France
Prof. Dragan Denić, Serbia
Dr. Ludmila Kuzmina, Russia
Prof. Željko Đurović, Serbia
Prof. Dimitri Lefebvre, France
Prof. Dragan Lazić, Serbia
Prof. Dragan Nešić, Australia
Prof. Zoran Ribar, Serbia
Prof. Zoran Pandilov, Macedonia
Prof. Goran S. Đordević, Serbia
Prof. Erik Noldus, Belgie
Prof. Ilija Nikolić, Serbia
Prof. Pavel Pakshin, Russia
Prof. Ljiljana Petković, Serbia
Prof. Dragoljub Pokrajac, USA
Prof. Ivan Milentijević, Serbia
Prof. Stefan Preitl,Romania
Prof. Leonid Stoimenov, Serbia
Prof. Emil Nikolov, Bulgaria
Prof. Dejan Rančić, Serbia
Prof. Alex M. Stankovic, USA
Prof. Gradimir Ilić, Serbia
Prof. Vesely Vojtech, Slovak Republic
Prof. Nenad Pavlović, Serbia
Prof. Xinghuo Yu, Australia
Prof. Milan Matijević, Serbia
Prof. Virginia Kiryakova, Bulgaria
Prof. Dragan Đurdanović, USA
Prof Radu-Emil Precup, Romania
Dr. Dragoljub Šurdilović, Germany
Prof. Ratko Pavlović, Serbia
Prof. Predrag Rajkovic, Serbia
Dr. Danijela Ristić-Durant, Germany
Prof. Žarko Ćojbašić, Serbia

Organizing Committee

Dr. Darko Mitić, president
Dr. Zoran Jovanović
Miloš Simonović
Marko Milojković
Vladislav Blagojević
Ivan Ćirić
Saša Nikolić
Ivan Pavlović
Dragan Jovanović
Staniša Perić

X Triennial International SAUM Conference
on Systems, Automatic Control and Measurements
Niš, Serbia, November 10th-12th, 2010

Pre-scheduled Control of Online Device-Based Backup and Incremental Data Backup in VMware Virtual System Environment

Vladimir Ćirić, Vladimir Simić, Ivan Milentijević

University of Nis, Faculty of Electronic Engineering, Department of Computer Science, Aleksandra Medvedeva 14, Niš, Serbia

e-mail: {vladimir.ciric, vladimir.simic, ivan.milentijevic}@elfak.ni.ac.rs

Abstract ***- The development of new virtualization platforms pushed the paradigm of backup systems a bit further then it was. Using virtualization online device-based backup can be considered as file-based backup, from hypervisor's perspective.The development and implementation of Linux script for pre-scheduled device-based backup, at one level, and incremental backup of user data on the other level, are proposed in this paper. Device-based backup is performed without interruption of user's activities. The backup process is illustrated by UML diagram. The proposed method is implemented on VMware platform using Linux script language. The system and user activities are monitored during the process of backup, and the results of the system resources usage are given.***

Keywords: ***Device-based backup, online backup, VMware virtual environment.***

I. INTRODUCTION

As stated in [1], near 70% of computer users, at least once have experienced a loss of important data. Data loss can be prevented by the price of creating backup copies of the data, at regular time intervals. The term *backup*, in information technologies, stands for an operation of creating a spare copy of the data, which can be used for data *restore* in cases of original data loss [2]. There are two main objectives when creating a backup copy. The first objective is to restore a complete state of all data as it was before the data loss. This process is well known as a "disaster recovery", where term *disaster* stands for natural disasters. The other objective is recovery of small number of files, mainly lost due to a user mistake [2].

There are a lot of techniques for backup copy creation. All techniques can be classified into major backup schemes: device-based scheme, and file-based scheme [3]. While number of device-based backup techniques don't allow system usage during backup process, there are techniques that don't interrupt user during device-based backup. These techniques are called *online* backup techniques [3]. User transparent backup is a development challenge since a beginning of backup software development, because of data consistency problem that exists with this particular backup technique [4].

Virtualization process can be considered as a layer of hardware or software resource abstraction, and it is first introduced in 1960s in the concept of virtual memory in IBM M44/44X computer [5]. Development of virtualization systems, which support virtualization of complete hardware of personal computers by representing a device with files, gives a new level of design freedom for device-based backups.

The aim of this paper is to describe development and implementation of system capable for pre-scheduled online device-based backup, as well as incremental data backup in virtual VMware environment [6]. The technique for solving data consistency problem during online device-based backup in virtual VMware environment will be proposed. User-transparent backup process will be illustrated by UML diagram. Proposed method will be implemented on VMware platform using Linux script language [7]. The system and user activities will be monitored during the process of backup, and the results of the system resources usage will be given. Having in mind resources requirements of online backup techniques, we will monitor and measure device-based backup separately from the file-based backup in VMware virtual environment, and measure user activities during both processes.

II. CLASSIFICATION OF EXISTING BACKUP TECHNIQUES

In order to clarify development of pre-scheduled control for data backup in virtual environment, we will briefly describe basic backup techniques in this section.

The simplest way to protect a file system against disk failures or file corruption is to copy the entire contents of the file system to a backup device. The resulting archive is called a full backup. If a file system is later lost due to a disk failure, it can be reconstructed from the full backup onto a replacement disk. Individually lost files can also be retrieved. Full backups have two disadvantages: reading and writing the entire file system is slow, and storing a copy of the file system consumes significant capacity on the backup medium [3].

Faster and smaller backups can be achieved using an incremental backup scheme, which copies only those files that have been created or modified since a previous backup [3, 4]. Incremental schemes reduce the size of backups, since only a small percentage of files change on a given day.

Physical or device-based backup systems ignore file structure when copying disk blocks onto the backup medium. This improves backup performance, since backup software performs fewer costly seek operations. However, this approach complicates and slows file restores, since files may not be stored contiguously on the backup medium [3].

While many backup programs require file system to remain quiescent during backup, on-line or active backup systems allow users to continue accessing files during backup. On-line backup systems offer higher availability, but introduce consistency problems. In [4] many of the difficulties with on-line backups when using programs like UNIX *tar* and *dump* commands are discussed. The most serious problems occur when directories are moved during backup, changing the file system hierarchy. Two strategies for overcoming these problems with on-line backup are locking and detection [3, 4].

Data consistency problem can be solved by creating a *system snapshot* in virtual environment [6]. In computer systems, a snapshot is a state of a system at particular point in time. When a device snapshot is taken in virtual environment, the state of the disk remains intact, and all changes are recorded in separate, so called *delta* file. This is a basic idea of online backup technique proposed in the paper, as it solves the data consistency problem, and reduces the complexity of online device-backup to the level of file-based backup.

III. SYSTEM AND SERVICE VIRTUALIZATION

In its conceived form, virtualization was better known in the 1960s as time sharing [8]. This technique would allow one programmer to develop a program on his console, while another programmer was debugging his, thus avoiding the usual wait for peripherals. Multi-programming, as well as several other groundbreaking ideas, begin to drive innovation. Two are considered a part of the evolutionary lineage of virtualization as we currently know it - the Atlas computer with concept of hardware supervisor, and IBM's M44/44X, with concept of virtual memory [5].

Concept of virtualization nowadays implies hardware resources abstraction, such that abstract hardware can execute complete operating systems. Physical computer which is responsible for virtualization is called *host*, while virtual systems are called *guest* computers, or guest operating systems. The *server virtualization* term stands for execution of multiple guest operating systems on single or multiple host system, while guest operating systems are not aware of existence of hosts beneath [5, 6].

One of the most critical components of a server virtualization platform is the hypervisor. A hypervisor is software that allows multiple virtual machines to share access to physical processor, memory, disk, and network resources. Examples of available hypervisors include VMware Workstation/Server, Microsoft Virtual PC/Server, and Parallels Workstation. Another type of hypervisor is sometimes called a bare-metal hypervisor [5]. Examples of bare-metal hypervisors include VMware ESX, Microsoft Hyper-V, and Citrix XenServer. The implementation of virtualization environment in this paper is performed by VMware ESX bare-metal hypervisor, which is based on Linux 2.6 kernel. System architecture is shown in figure 1.

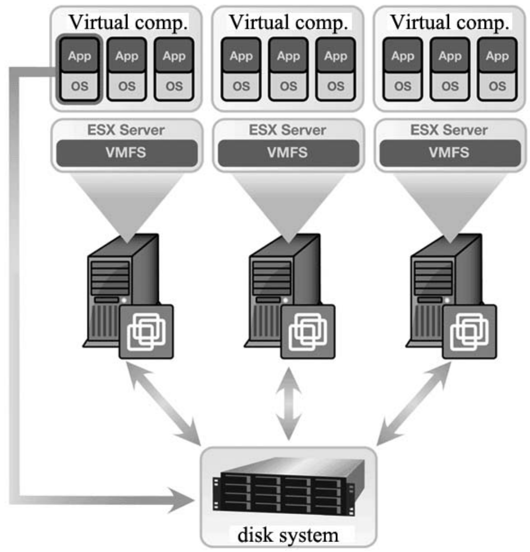

Fig. 1. Architecture of VMware ESX bare-metal virtualization environment

As ESX hypervisor sees a virtual machine, virtual server is a set of files on physical disk, in directory allocated to the virtual server. Typical file listing of virtual server directory is shown in Fig. 2.

Files that have *.vmdk* extension represent virtual disks of virtual server, *.vmsd* files hold info about system snapshot, *nvram* is virtual BIOS, *vswp* virtual memory of virtual computer, and *vmx* is configuration of virtual computer that holds all infos about virtual resources available to the guest [6].

```
total 21G
-rw------- 2.8G Mar 30 21:02 L3public-3b4006ff.vswp
-rw------- 15G Apr 26 19:04 L3public-flat.vmdk
-rw------- 8.5K Apr 26 06:49 L3public.nvram
-rw------- 575 Apr 26 06:49 L3public.vmdk
-rw------- 862 Apr 26 06:03 L3public.vmsd
-rwxr-xr-x 2.6K Apr 26 06:49 L3public.vmx
-rw------- 263 Mar 22 14:38 L3public.vmxf
```

Fig. 2. Typical files that represent virtual computer on ESX hypervisor

IV. DEVICE BACKUP IN VIRTUAL ENVIRONMENT

Online device-based backup process complexity can be reduced to file-based backup if virtual server is observed from ESX's point of view (Fig. 2).

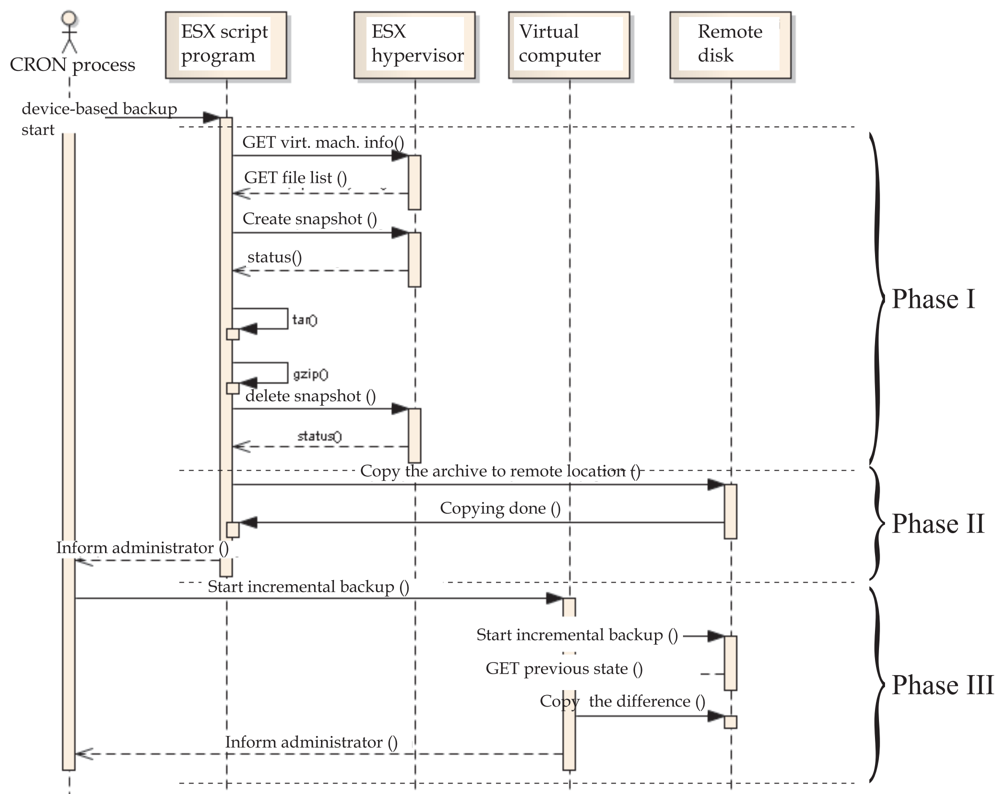

Fig. 3. UML diagram of command sequence that controls execution of online device-based backup in VMWare ESX environment: Phase I - snapshot and archiving the files; Phase II - copying to remote location; Phase III - incremental backup

Data consistency problem in this environment can be eliminated by creating a snapshot of virtual machine. In that way the virtual server can be exploited by users, while ESX is creating a backup of snapshotted virtual disks and the other resources, except the delta file that records current user activities. VMware's ESX offers a Linux script language that has a support for virtual machine control [7], which makes ESX hypervisor a suitable environment for pre-scheduled snapshotting and archiving the snapshotted virtual machines. As a commercial virtualization solution, VMware offers solution called VMware Consolidate Backup. In contrast to proposed technique, VMware Consolidate Backup requires existence of dedicated computer to control the process [6].

The technique proposed in the paper doesn't require dedicated computer to control backup process. Instead of that, it exploits ESX hypervisor scripting capabilities to schedule and control backups. The sequence diagram of operations during the backup process is shown in Fig. 3. There are several objects involved in backup process: Linux Cron process, which schedules execution, ESX script, used to control virtual machines, hypervisor, virtual machine, which is the object of the backup, and hard disk for the remote backup (Fig. 3).

The backup process is divided into three phases. Pre-scheduled execution is achieved by using Linux Cron process, which executes ESX script in given time intervals (Fig. 3). The first operation, executed in the first phase, right after Cron initiates the backup is done by developed ESX script program (Fig. 3). The developed script program checks the virtual machine, and if the machine is powered on the script invokes ESX procedure to create a machine snapshot. Virtual machine control is provided by ESX's Linux shell command *vmware-cmd* [7]. Hypervisor then snapshots the machine by creating a separate *delta* file in virtual machine's directory. From that instance all user activities are written in the delta file, so original disk state remains intact. This enables user transparency of the proposed technique for online device-based backup.

Next action done by the developed script is packing of all relevant files into Linux *tar* archive, and compressing the archive by Linux *gzip*. After the compressed archive is created, the snapshot is removed, and buffered user actions are applied into the virtual machine, which ends the first backup phase (Fig. 3). The second phase is devoted to storing the compressed backup archive that holds a snapshot of the virtual machine to a remote disk (Fig. 3).

As creating a full backup copy of a device is resource consuming operation, from both CPU/memory perspective and network perspective frequent full backup isn't recommended. This is the reason for involving the third phase in the sequence diagram in Fig. 3. The third phase stands for incremental backup of small number of files that have been changed between two full backups.

V. SYSTEM IMPLEMENTATION

The proposed system is implemented on three IBM x3550 servers, equipped by IBM DS3000 DataStorage, which stores virtual servers. The system's architecture is shown in Fig. 1. System is equipped with additional network disk, on remote location, which is used for storing archived snapshots of virtual machines.

The developed ESX script program, which implements the sequence diagram from Fig. 3 is scheduled by Linux Cron, and executed by ESX hypervisor on the servers. The archive copying from phase II (Fig. 3) is implemented through dedicated SSH tunnel between the hosts [5]. Full device-based backup is initiated once in three days, while incremental backup is performed twice per day.

Resources usage of both ESX server and virtual machine, which is backed-up, are given in Fig. 4. Fig. 4a, b and c show usage of ESX's CPU, disk and network resources, while Fig. 4d, e and f stand for the virtual machine. The virtual machine shown in the figure is used, among the others, for laboratory exercises of Algorithms and programming course at Faculty of Electronic Engineering, University of Nis, Serbia, and in the moment when resources usage from Fig. 4 is recorded there were more than 60 users logged into the course system.

In Fig. 4 it can be seen that in the Phase I there is high processor and disk resources usage of ESX server, while in the Phase II the network activity is higher then average. In Fig. 4d it can be seen that there is user activity during the Phases I and II. The third phase in Fig. 4 reflects activities during incremental backup, and shows increased virtual disk usage, due to the search of changed files for incremental backup. The network activity in this phase is lower than network activity of the users.

Regardless of lack of requirement for dedicated computer, usage of the technique proposed in this paper is justified in the cases when ESX server has enough resources to perform a backup during a regular system load.

VI CONCLUDING REMARKS

The method for pre-scheduled online device-based backup in virtual environment is proposed in this paper. The proposed backup technique doesn't interrupt the regular user activities. Due to higher resource usage during full device-based backup, the technique offers more frequent incremental backups. The proposed method is implemented on VMware platform. The implementation of proposed backup technique is embedded into the script program, and can be easily deployed to existing virtual environment. The price paid for the backup is increased ESX server resources usage. The backup technique is illustrated by UML diagram. The system and user activities are monitored during the backup, and the results of the system resources usage are presented.

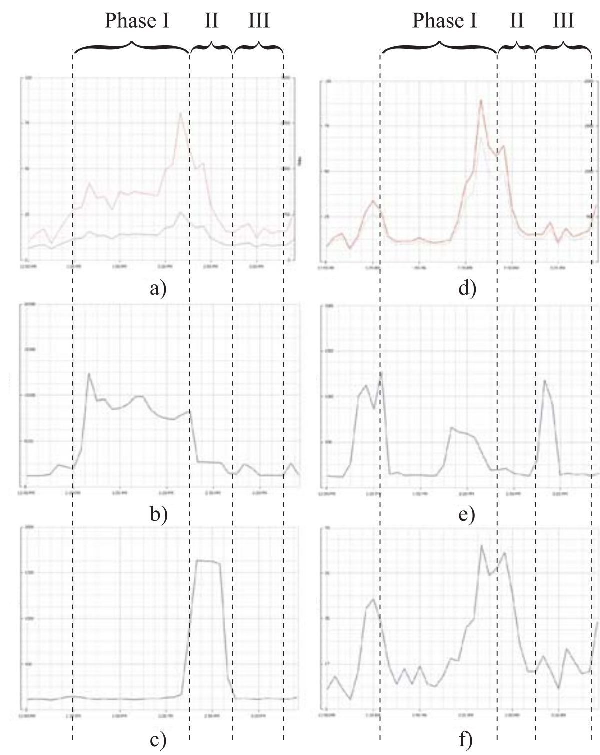

Fig. 4. System's resource usage during the backup: a), b) and c) - CPU, disk, and ESX server's network, d), e) and f) - CPU, disk, and virtual machine's network

REFERENCES

[1] W. Curtis Preston, "*Backup & Recovery*", O'Reilly, US, 2007.

[2] Preston de Guise, "*Enterprise Systems Backup and Recovery: A Corporate Insurance Policy*", CRC Press, 2009.

[3] Ann Chervenak , Vivekanand Vellanki , Zachary Kurmas, "Protecting File Systems: A Survey of Backup Techniques", *Joint NASA and IEEE Mass Storage Conference*, March 1998.

[4] S. Shumway, "Issues in On-line Backup", *In USENIX Proceedings of the 5th Conference on Large Installation Systems Administration*, September 1991, pp. 81-88.

[5] John Hoopes, et al., "*Virtualization for Security*", Elsevier, Inc., 2009.

[6] Al Muller, Seburn Wilson, "*Virtualization with VMware ESX Server*", Syngress Publishing Inc., US, 2005.

[7] Al Muller, et al., "*Scripting VMware*", Syngress Publishing Inc., Canada, 2006.

[8] Christopher Strachey, "Time Sharing in Large, Fast Computers", *International Conference on Information Processing*, June 1959, pp. 336-341.