THE DANISH BUSINESS ACADEMY DANIA - VIBORG

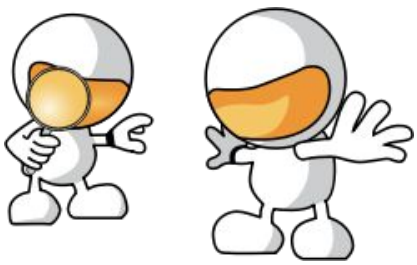
FINAL PROJECT

FREE AUTOMATED BACKUP SOLUTIONS FOR FREE ESXi VIRTUAL SERVER

Prepared by Radu Gabriel Popa • May 15, 2015

Guidance Teacher - Paul Saunders

Name of the company



Taycor Financial

ITT 2013

The number of characters(with spaces): 44678

This task is not confidential

1.Introduction.

My internship takes place at Taycor Financial. Taycor is a small bank located in Los Angeles - California, with main activity in leasing equipment for companies in many different fields of activities.

On my first day I was introduced to some of the projects that the company had already started and they would like to implement as soon as possible.

The project with the highest priority is the one I have chosen to write about in my Final Project: find a solution to make a backup of the database and the CRM software. This should be done automatically, without entrapping or stopping the servers.

2. Problem statement.

Building an automated backup solution for company's database and Customer Resources Management Software.

Currently the company has 1 Windows Server 2003 and 1 Windows SQL Server 2005, running on top of a VMWare - ESXi, virtual server. The CTO manages them over the Vsphere client. Considering that the CRM and the database servers have to handle the traffic from 10 persons, the current performances of the network infrastructure is high.

The president of the company plans to grow the number of employees with 50% by the end of the year. In order to implement this growth, there is a need of some software testing that can perturb the normal work flow of the current employees.

The reason why there is not yet a backup solution is because there is only one technology officer that works on many projects.

In order to find a reliable solution someone has to study how to backup a virtual server and run some tests outside the normal working hours, namely after 6 PM.

I decided that this is a project that I could work on because there is not a tight schedule. When I will have difficulties I can always ask the advice of the company's technical consultant.

My supervisor established as a target and main challenge to find a solution that will not add any licensing costs, namely open source solution.

In order to solve this challenge I will have to understand what is virtual server and how it works.

I will test all the major backup software (assuming there are already some third party software) and will analyze Graphical User Interfaces and Command

Line Interfaces software. I will look at aspects like, reliability, speed of backup, file integrity, ease of use and most of all **licensing cost**.

3. What is ESXi virtual server ?

3.1. Why virtualization ?

Physical servers require space, electrical power and cooling. We end-up spending quite a bit of time doing lifecycle management for those servers in terms of aligning ourselves with warranty coverage and replacing those servers before the warranty coverage becomes expired. Server lifecycle rarely matches application lifecycle. Older operating systems may no longer be supported. Older applications may not support newer operating systems.

When we move into a virtualized environment we can take a lot of those application that may have run on a physical server in order to maintain their separate configuration environment, but they may not be using the resources of that server effectively. We can consolidate those services into virtual servers running on physical servers but at a higher consolidation ratio. So that may potentially reduce the number of physical servers that we have on our environment that's gonna help reduce our power consumption and cooling footprint and overall move to a more green data-center, more efficient and more dynamic then we have typically to deal with before. It will also help up with testing and deployment cycles because we can provide a test environment much more easily.

3.2 Hypervisors.

A hypervisor provides a layer between a hardware and a virtual machine. It is not an emulator or simulator. Vmware supports only x86 / x64 on x64 hosts with an excellent performance.

There are 2 types of Hypervisors:

Type 1: **Bare Metal,** named **ESXi** - it interacts directly with the hardware. This type of hypervisor it requires native drivers and provides official support for specific physical hardware.

Type 2: **Application/Service,** named **Workstation,** - runs as an application – service on another host Operating System and it relies on that OS’s drivers.

3.3 Virtual Machines.

The virtual machines use virtual hardware and run standard x86/x64 OS. Vmware provides a specific set of virtual hardware:

- One BIOS – acronym from Basic Input/Output System, is a combination of a hardware device and computer instructions and data that reside as read only software.
- One CPU/chipset - *"is a set of electronic components in an integrated circuit that manages the data flow between the processor, memory and peripherals. It is usually found on the motherboard. Chipsets are usually designed to work with a specific family of microprocessors. Because it controls communications between the processor and external devices, the chipset plays a crucial role in determining system performance"*. WIKIPEDIA.
- 2 types of NIC – stands for "network card" or "Network Interface Controller". It is a computer hardware component that connects a machine to a network.
- 2 types of SCSI adapters - "Small Computer System Interface", *is a set of standards for physically connecting and transferring data between computers and peripheral devices. SCSI is most commonly used for hard disks and tape drives, but it can connect a wide range of other devices, including scanners and CD drives.*

  WIKIPEDIA.

The guest Operating System must have driver support for these virtual devices. Vmware tools provides a number of special divers to support advanced features such as memory demand management.

3.4. ESXi Server - Hardware Limitations.

For ESXI 5.1, the version that I am working with, the maximum hardware features that the hypervisor supports are the following:

- Number of virtual CPUs per host 2048;
- Number of logical host CPUs 160;
- Maximum host RAM 2 TB;
- Virtual machines per host 512;
- Virtual disks per host 2048;

3.5. Virtual Machine Limitations.

A virtual machine can have:

- Maximum RAM 1TB;
- Maximum virtual CPUs 64;
- Maximum virtual disk size ~2TB;
- Number of virtual disks per virtual machine 60.

3.6. VMware ESXi.

Replaces the previous Vmware ESX bare metal hypervisor. Designed to reduce the footprint of the installed Operating System that the Virtual Machines

operates with. ESX it was pretty much a full scale Linux Distribution with quite a few packages and utilities installed that were not necessary commonly used in the Vmware environment, but were provided to make it easier to support the Linux style environment that Linux administrators were familiar with. That lead to an expanded disk footprint and security issues. So a lot of that were reduced to bare minimum. There is still a minimum Linux environment for core system management and advanced troubleshooting. ESXi does not support running Linux tools, applications or management agents.

There are 2 ways to connect and manage an ESXi server: ESXi Shell Commands and vSphere Client.

3.7. ESXi Shell Commands.

In order to use ESXi Shell Commands first thing is to enable SSH access from DCUI interface that comes as standard interface after installing ESXi 5.1.

Image 1 - ESXi 5.1 DCUI Interface

To connect to an ESXi host using SSH I can use any SSH client. I will use "Putty". I need to know the server's IP, log-in user-name and password.

After connecting I can start typing Linux commands

Image 2 - Initiating connection from Puty

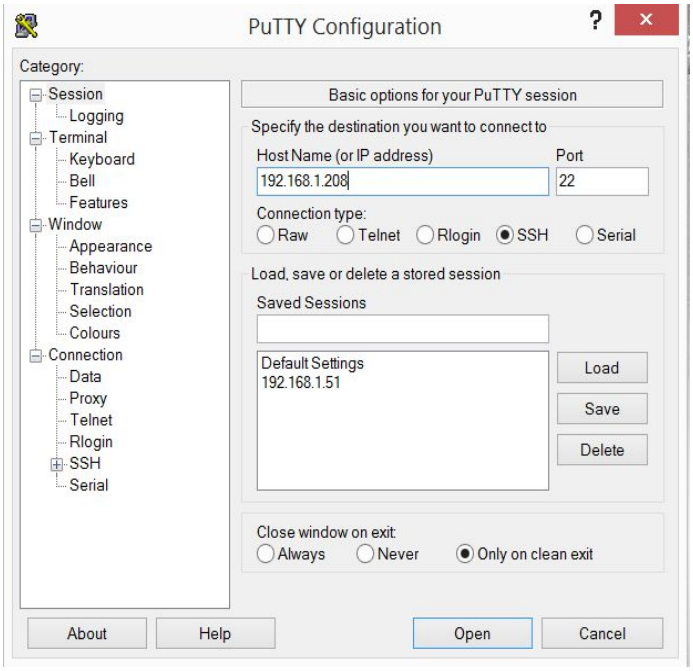
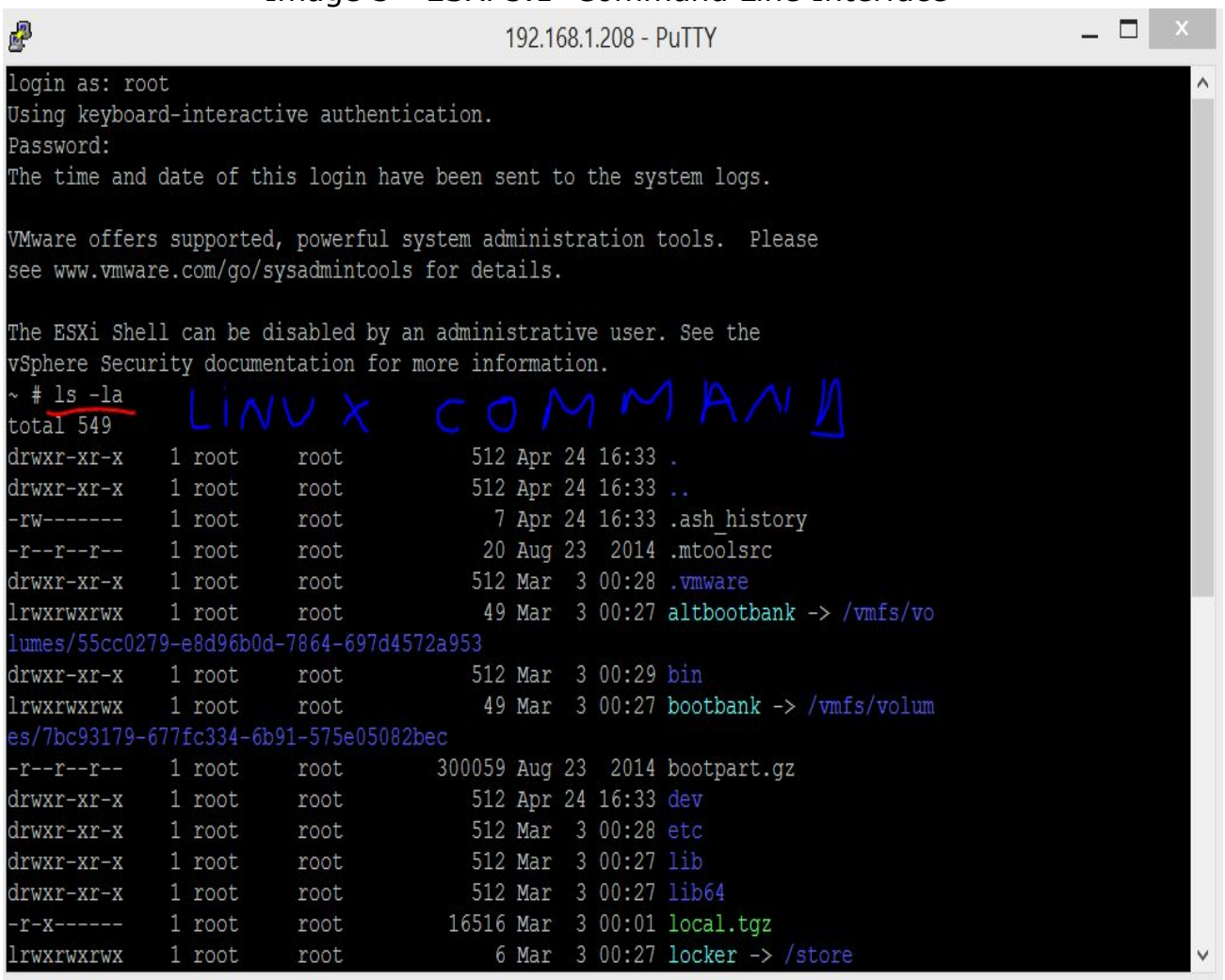


Image 3 - ESXi 5.1 Command Line Interface

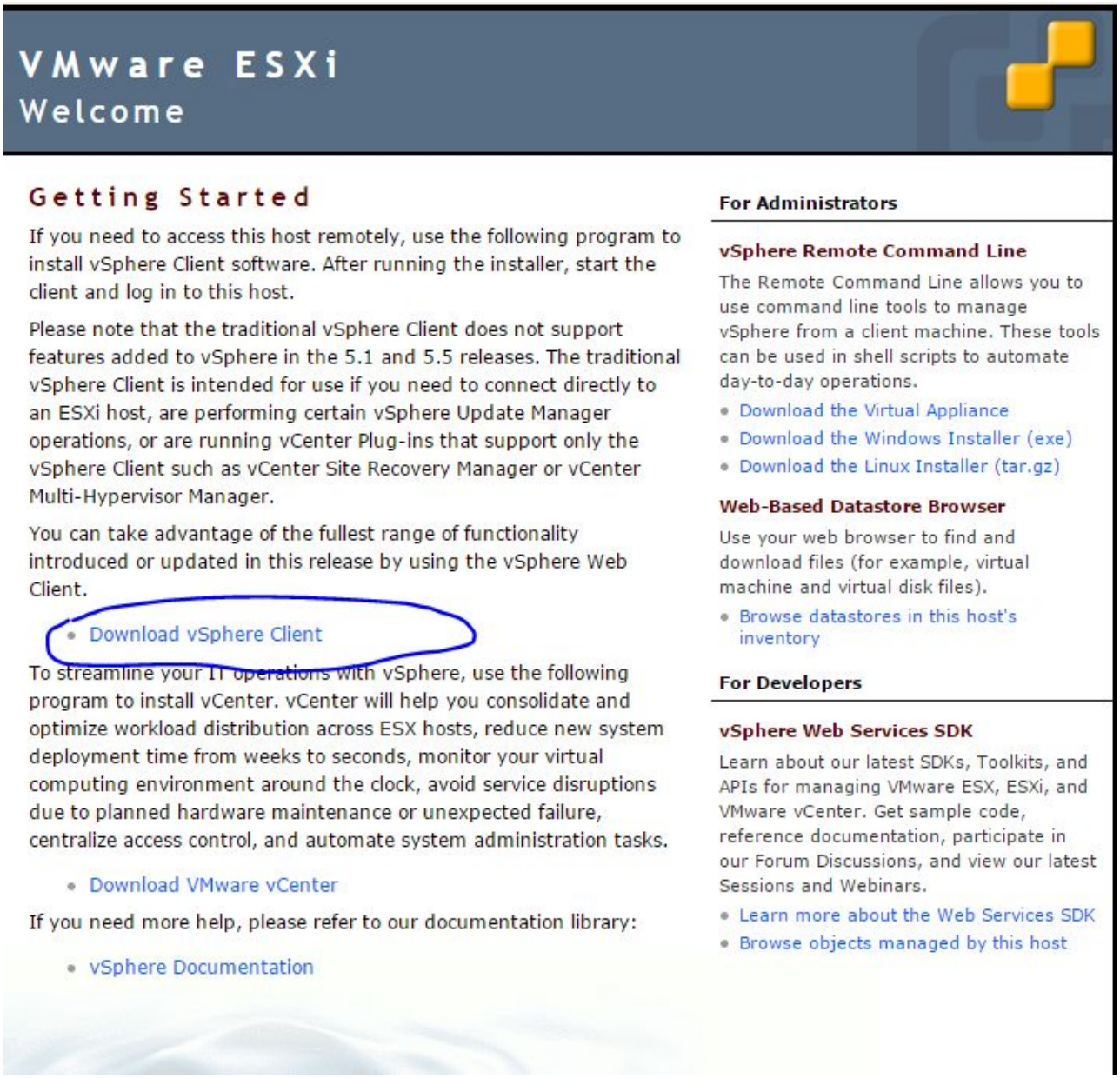


3.8 vSphere.

VSphere is a virtualization platform that enables the IT Administrator to interact with the ESXi server.

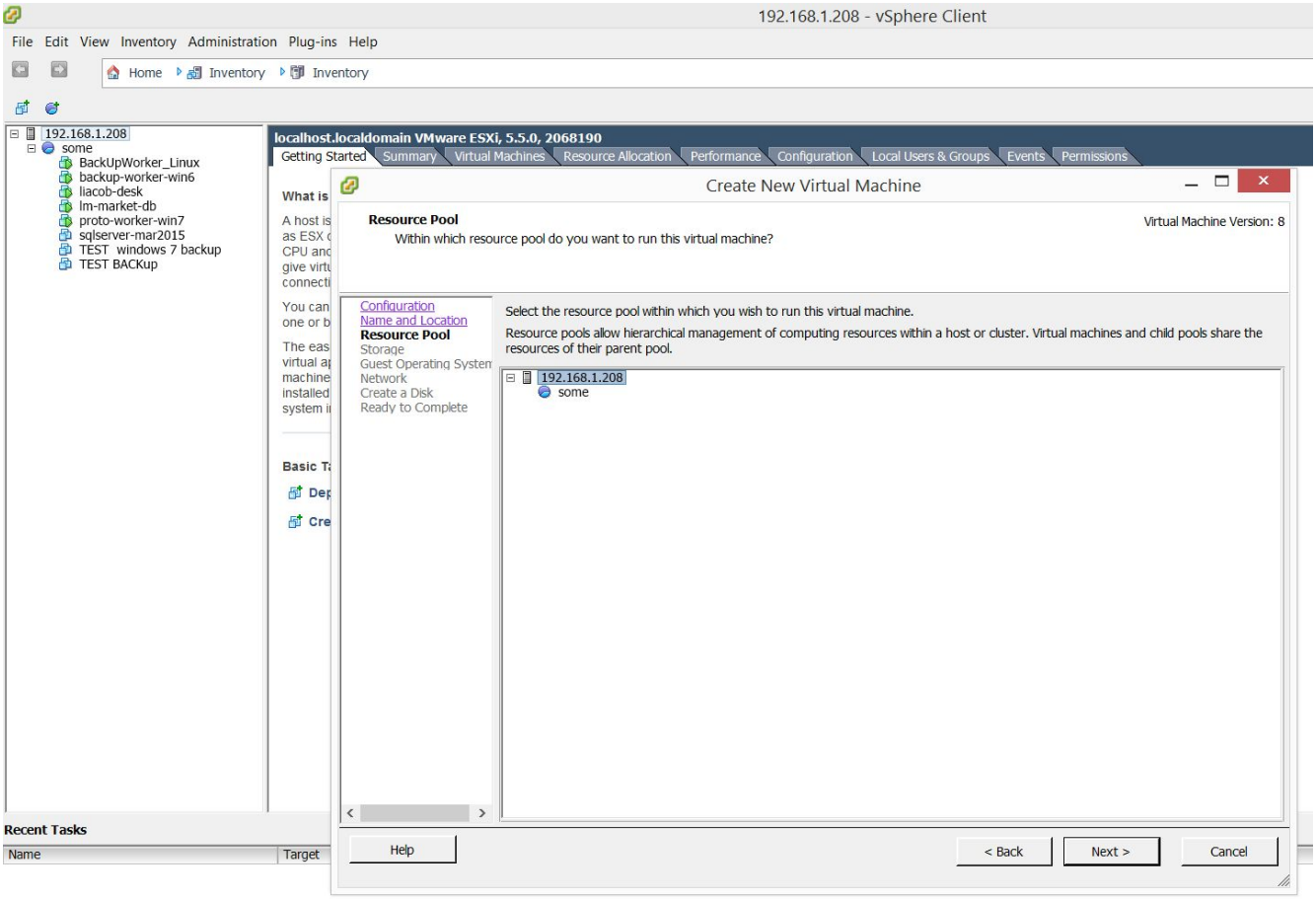
Installing the vSphere Client is just as simple as installing any other Windows software: Next → Next → Finish. In order to download the Windows Installation Kit, I have open any web browser, my favorite is Google Chrome, and type in the server's IP address "https://192.168.1.208/". I will receive a security notification from the browser telling me that he doesn't know the specified IP address is safe. To skip this notification I have to click Advanced and "Proceed to 192.168.1.208 (unsafe)" and click "Download vSphere Client"

Image 4 - Web browser Welcome Page



After the installation is done, I'll have a nice Graphical User Interface Software that is allowing me to create new virtual machines. The process of **creating new VMs** is fairly easy and intuitive: File → New → Virtual Machine. Then practically I have to define the new "Virtual Hardware Configuration" for the new Virtual Machine. Those properties I can determine by reading the minimum requirements for the new Operating System that I will install.

Image 5 - vSphere 5.1 Client after login



4. Understanding what data is most important to backup ?

First of all, what means "to do a backup ?"

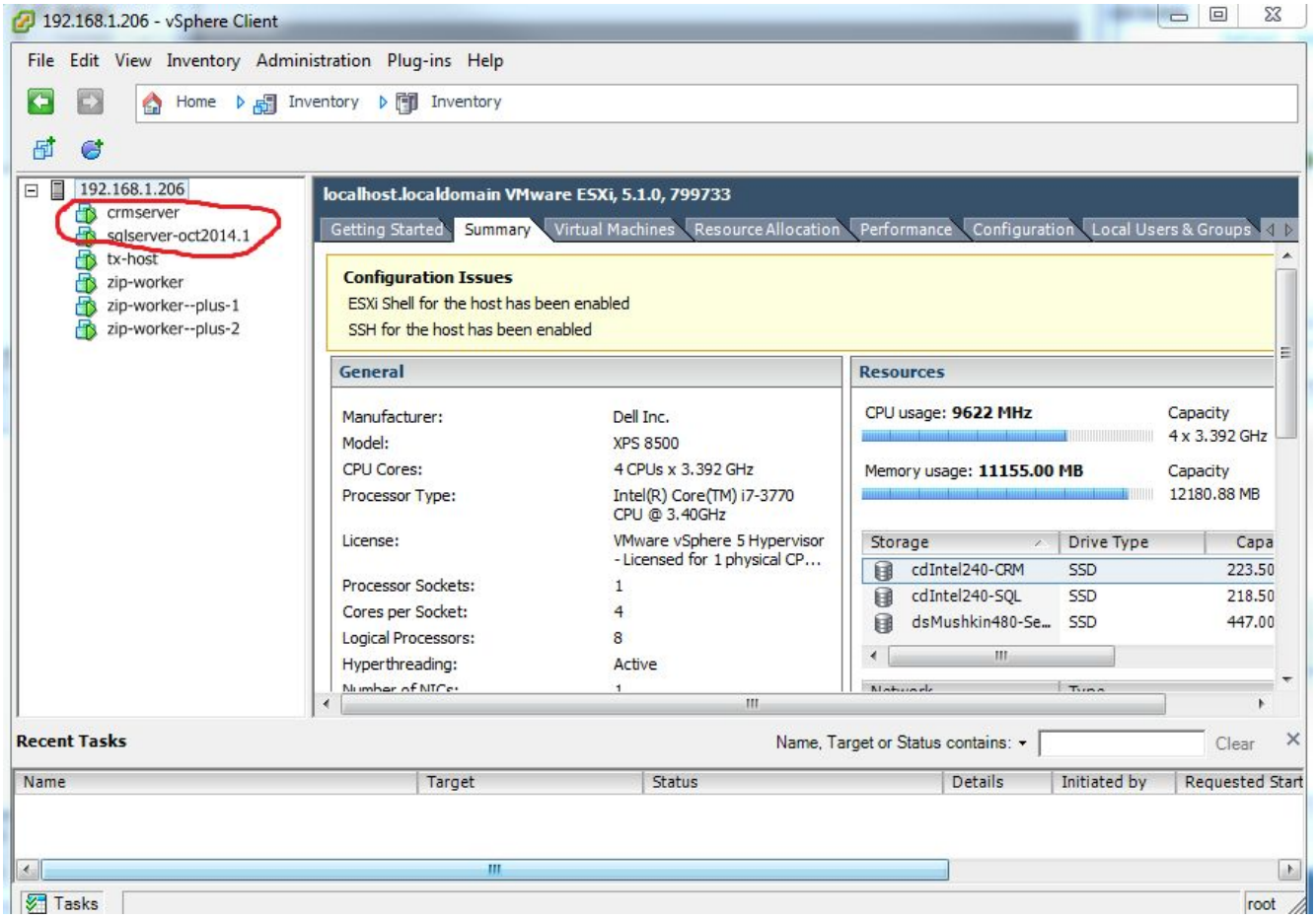
*"In information technology, a **backup**, or the process of backing up, refers to the copying and archiving of computer data so it may be used to **restore** the original after a data loss event. The verb form is to back up in two words, whereas the noun is backup."* WIKIPEDIA.

In short terms, "Taycor Financial" sales financial services to other companies. The main work is done by the sales representatives and account managers. Their work is done 100% in Sage CRM. Without this software the company

practically stops working. Sage CRM runs on Windows Server 2003 Virtual Machine. The CRM is connected with a 2005 Windows SQL Server, also running on Windows Server 2003.

So, basically, in case of a disaster event happens, 2 virtual machines must be fully backed up: **"crmserver"** and **"sqlserver-oct2014.1"** (see picture below)

Image 6 - Screenshot from main ESXi server



5. How to backup a running virtual machine ?

According to the VMware official ***"Best practices when backing up a VMware Workstation virtual machine"*** *:*

Best practices when backing up a VMware Workstation virtual machine(2006202)

Purpose

This article describes:

- *Best practices for backing up a VMware Workstation virtual machine.*
- *Issues to consider when backing up a VMware Workstation virtual machine.*
- *Steps to back up a VMware Workstation virtual machine.*

Resolution

To back up the virtual machine (which includes the operating system, application files, settings, and user data), you need to make a copy of the folder in which the virtual machine is stored. Ensure that the virtual machine is not running or suspended while you are backing up the virtual machine.

To back up the virtual machine:

1. *Ensure your virtual machine is in a powered off state.*
2. *Locate the virtual machine folder. For more information, see Locating a hosted virtual machine's files (1003880)*
3. *Right-click the virtual machine folder and click **Copy**.*
4. *Navigate to the folder in which you want to store the backup, right-click anywhere within the folder, and click **Paste**.*

***Note**: Replace the backup virtual machine on a regular basis using the process described above.*

You can:

- *Choose to move the backed up virtual machine to an external storage device, such as a flash drive or USB hard drive.*
- *Create a copy of the virtual machine on your desktop, then copy it to DVD or BD.*
- *Keep the snapshots of the virtual machine.*
-
- ***Note**: Before beginning the back up, you can commit the snapshots to save storage space.*

My objective is to back up **running** virtual machines. In order to "freeze" a virtual machine at a particular moment in time a "snapshot" must be created. Then in the future I will have the option to revert to that "snapshot" moment.

When taking a snapshot of a virtual machine there 2 options that I can enable:

- "Snapshot the virtual machine's memory"
- "Quiesce guest file system".

After doing some restore tests I got to the conclusion that if that I enable either one of the 2 options some applications and software may crash easily.

Not enabling both options the data that was written permanently to the virtual disk will be saved, meaning that all database files will be saved. Just currently working files will not be saved, minor data loss that it is acceptable by my supervisor. Also the speed of the snapshot is very fast by capturing just the files written to the virtual disk.

6. Starting point : what type of license the company has for ESXi ?

The answer to this question is very short and was provided by my IT supervisor: *"We're using a registered versions of the VMware ESXi Free product".*

Then I start searching more details about VMWare products.

Here's my understanding of their vSphere offerings:

- VMware ESXi Free (requires registration and a license to use for more than 60 days in trial) – only comes with the basic vSphere client with no API access.
- VMware vSphere Essentials $260/year – comes with the vCenter product (a web based vSphere client) and **unlocks API access**.
- VMware vSphere Essentials Plus $2,740/year – adds some advanced modules to vCenter that facilitates things like live-backups and implied support

Image 7 - Features comparison between vSphere editions

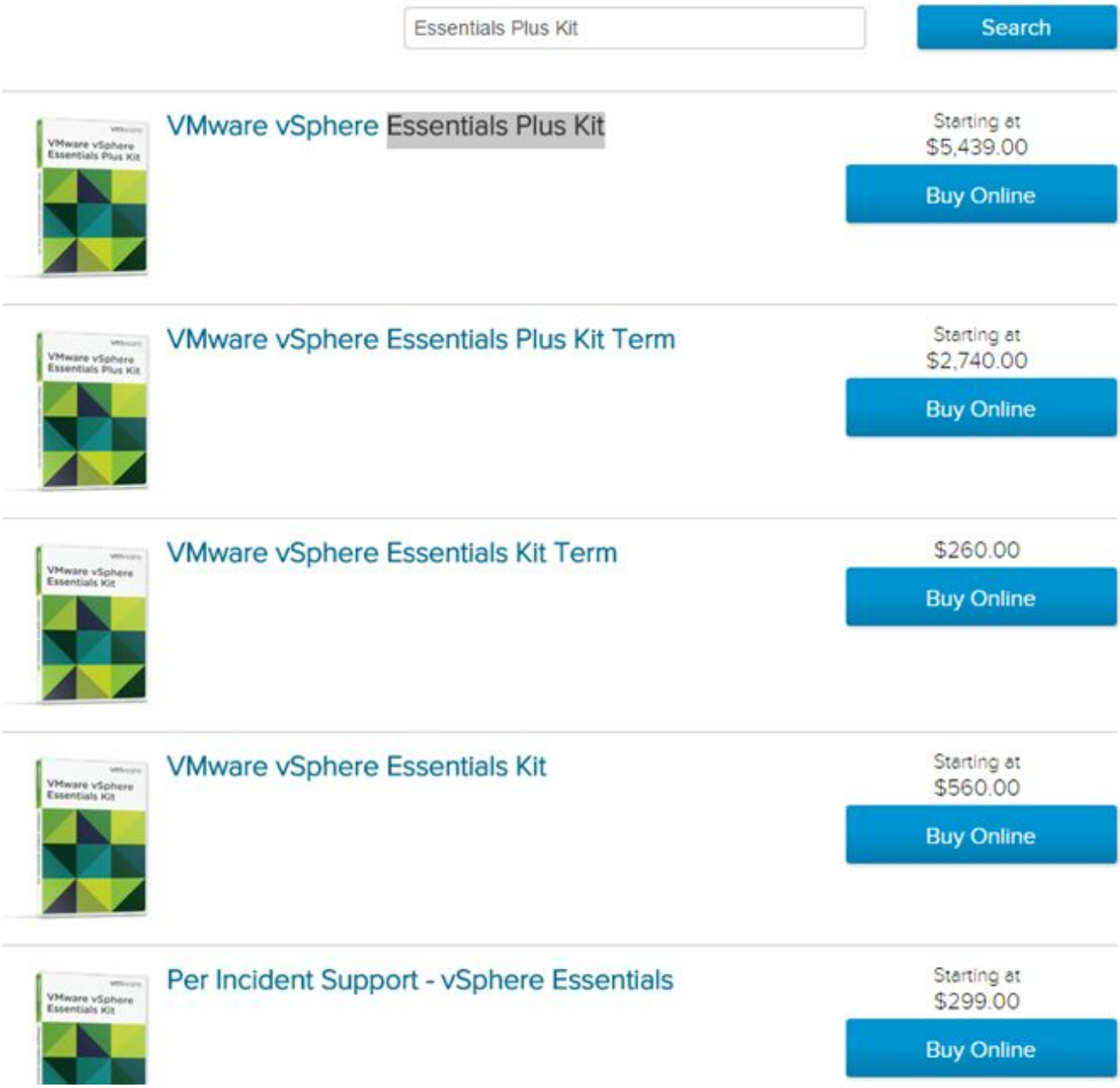
data protection

VMware vSphere Essentials Kit Term Details

Benefits | What's Included | Compare Kits | Resources

Product Overview	Essentials Kit	Essentials Kit Term	Essentials Plus Kit	Essentials Plus Kit Term
Overview	Server virtualization and consolidation with centralized management	Server virtualization and consolidation with centralized management	Server virtualization and consolidation plus business continuity	Server virtualization and consolidation plus business continuity
Centralized Management	vCenter Server Essentials	vCenter Server Essentials	vCenter Server Essentials	vCenter Server Essentials
License entitlement	3 servers with up to 2 processors each	3 servers with up to 2 processors each	3 servers with up to 2 processors each	3 servers with up to 2 processors each
Features	vSphere Hypervisor	vSphere Hypervisor	vSphere Hypervisor, vMotion, Cross Switch vMotion, High Availability, Data Protection, vShield Endpoint, vSphere Replication	vSphere Hypervisor, vMotion, High Availability, Data Protection, vShield Endpoint, vSphere Replication
License Duration	Perpetual with updates for the selected subscription term	1 Year (Renewable)	Perpetual with updates for the selected subscription term	1 Year (Renewable)

Image 8 - Pricing comparison between vSphere editions



"Data protection" is the feature offered by VMware as a backup solution.

So, official backup support for ESXi 5.1 from VMware company starts at $2740/year. To unlock APIs access $260/year are required. Any third party backup software like VEEAM, require at least access to APIs.

7. Doing a manual backup of a Virtual Machine.

Playing with the company's virtual server it is a risky task, so my supervisor decided they can buy another physical server with good performance hardware. This will allow me to test future backup solutions that I will find. After replicating the 2 virtual machines also we can develop different applications to integrate them with the CRM.

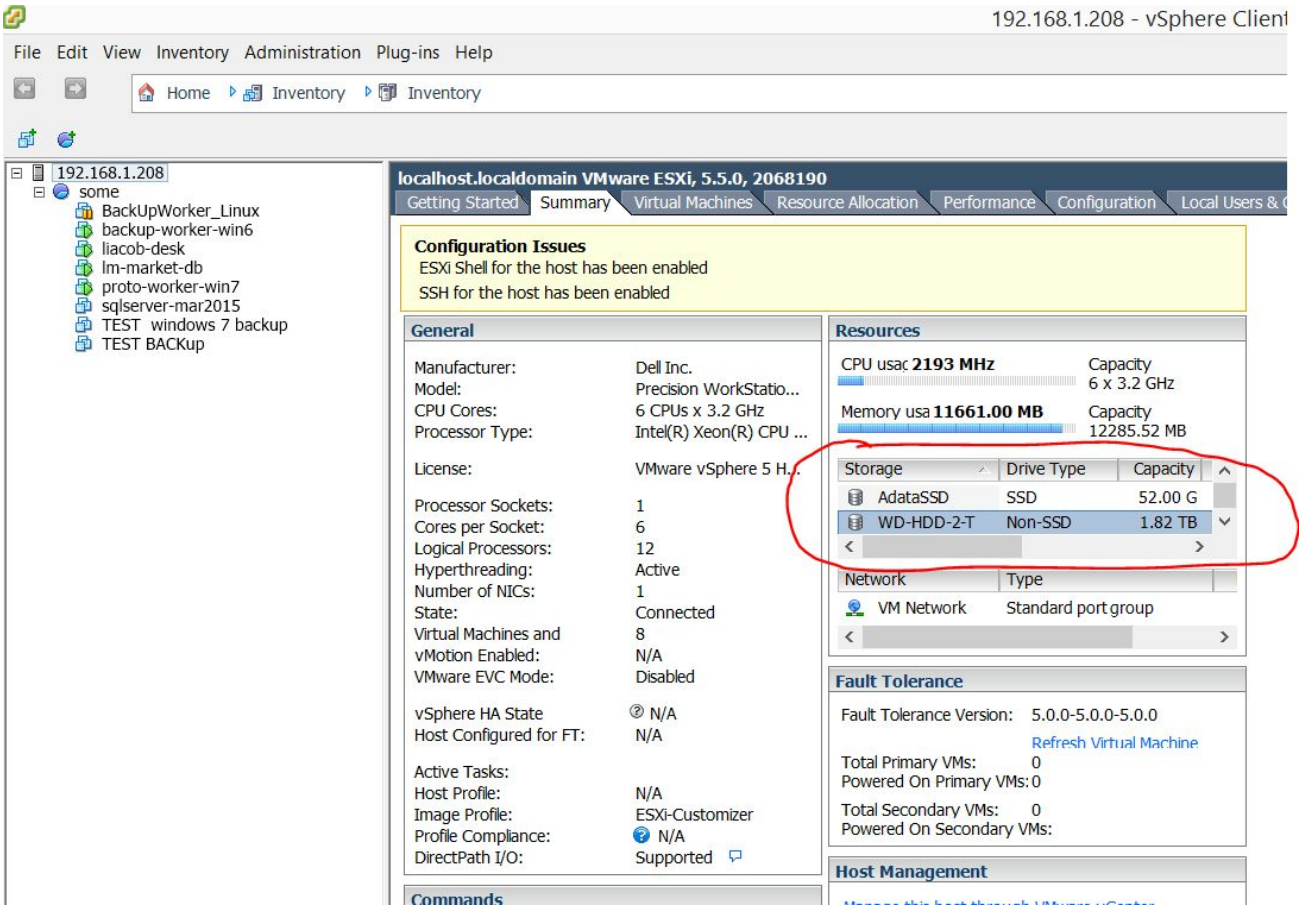
Now I have to opportunity to install for my first time a **Bare Metal Hypervisor** on a physical machine. After that I installed some virtual machines different types and Operating Systems: Windows XP, Windows 7 and Linux Mint.

I found out that I can access the folders where the virtual machines are located using vSphere: Summary tab → Under Storage I can see the physical hard drive's name → right click → browse datastore.

The fastest way to do a backup is to copy the files and folders on the physical machine. So we decided to add a second hard disk to the server and back up the 2 virtual machines following the VMware best practices recommendations.

For testing purpose I have one fast 55 GB Adata SSD that the ESXi Server runs on and a classic 2TB Western Digital hard drive for backup.

Image 9 - Storage, physical disks available



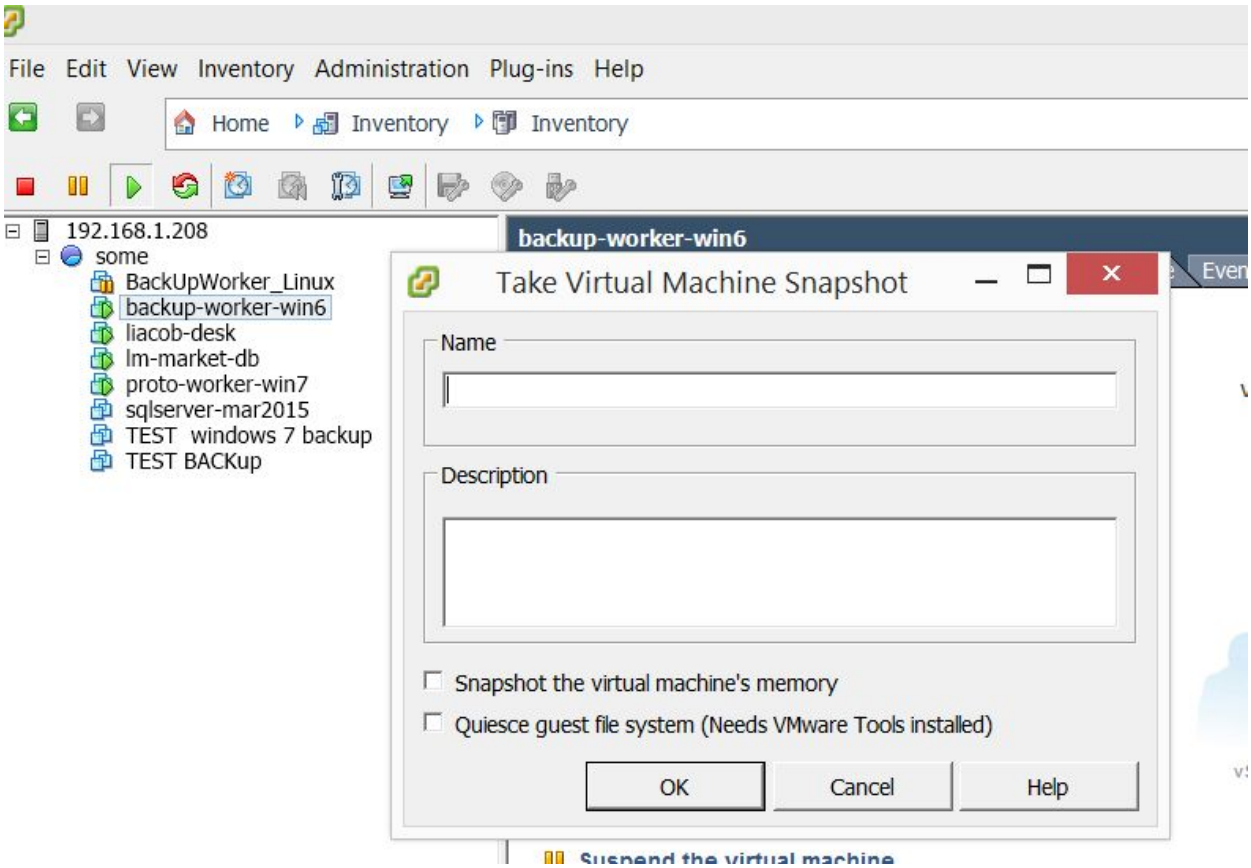
Before copying folders where the virtual machines are stored from one physical hard drive to another there is one very important aspect I must mention.

Don't forget to take the snapshot of the Virtual Machines !!!

To do this just right click on the virtual machine's name → select Snapshot → Take Snapshot. This will pop out a small window where I have to give a name

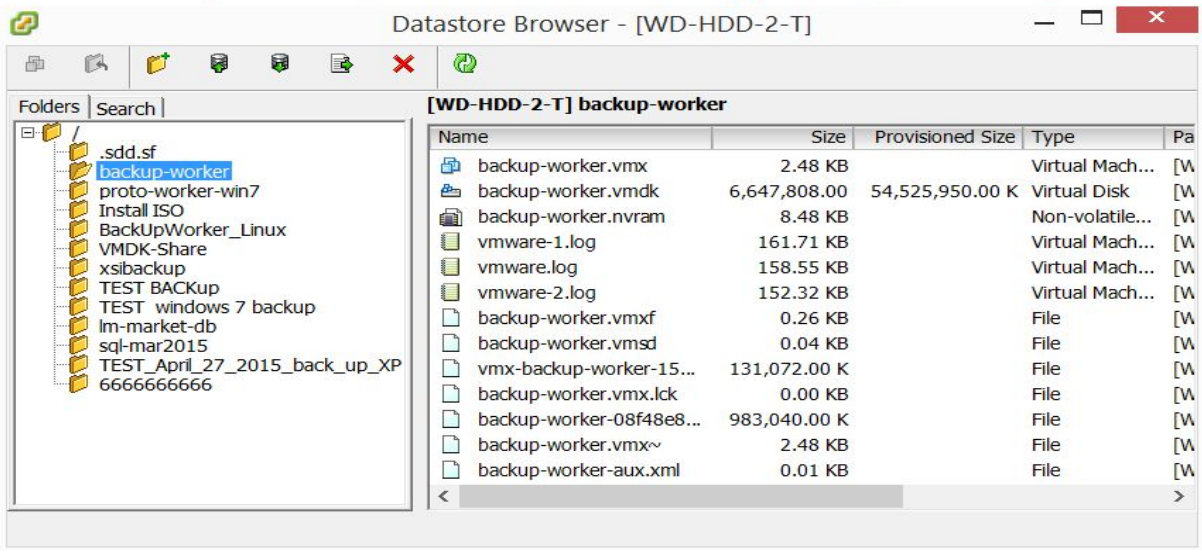
for the Snapshot and uncheck the 2 options so the process of "snapshotting" is very fast.

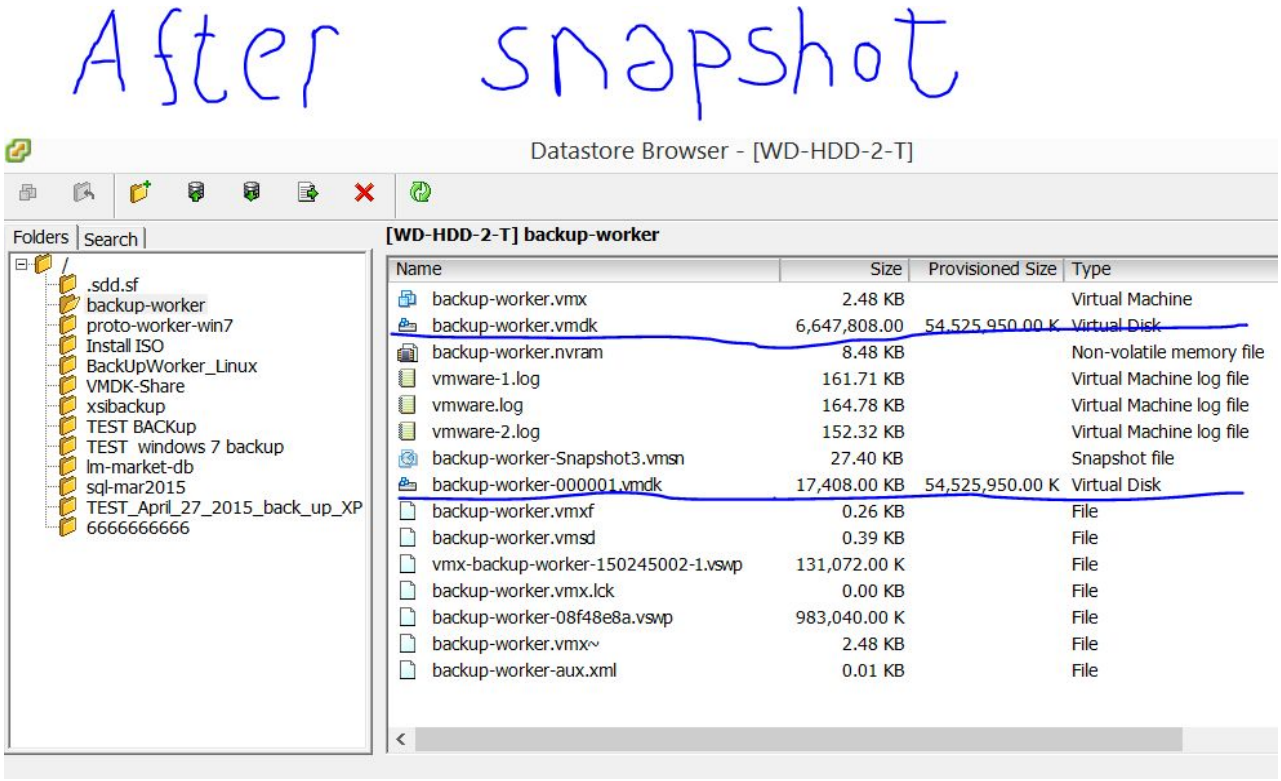
Image 10 - Taking a snapshot of a virtual machine



Important Notice: after the snapshot is taken a new virtual disk file is created and the virtual machine starts writing to that virtual disk allowing me to copy the normal virtual disk file "backup-worker.vmdk" , the virtual machine configuration file "backup-worker.vmx" and some other files in that folder.

Before snapshot





Even if not all files were copied successfully because they were used by the virtual machine, having the virtual disk file "backup-worker.vmdk" and the virtual machine "backup-worker.vmx" is enough to restore the virtual machines.

The speed of the backup process equivalents with the speed of copying the content of the folder "backup-worker". According to the Performance Chart tab from vSphere the average speed was about 5 MBps.

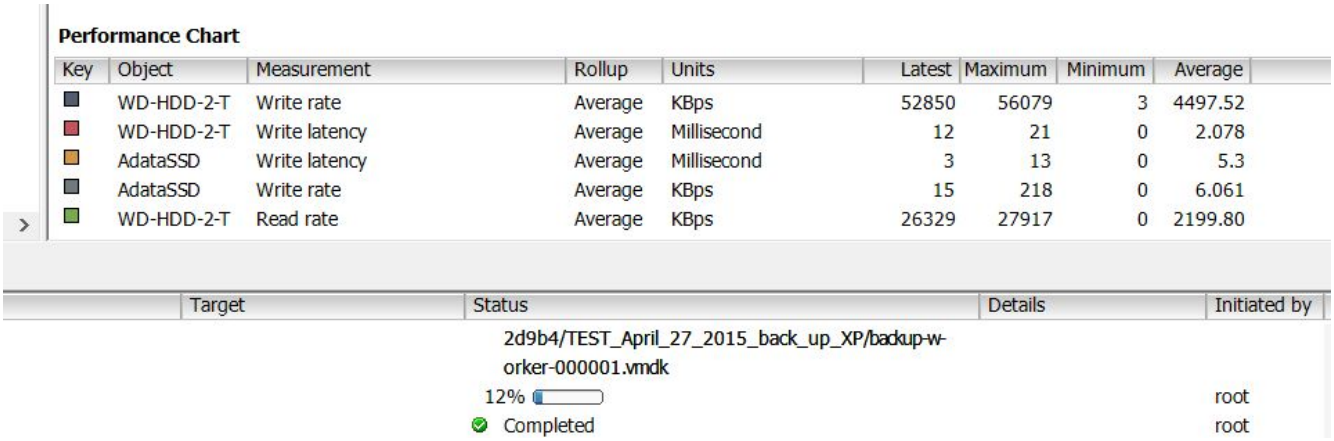
The whole process lasted 4 minutes and the size of the folder content was 6.65 GB –-> 6650 MB \ 2400 s (seconds) = 2.77 MBps.

So the **real backup speed** is **2.77 MBps.**

I must mention that this speed was achieved by copying one folder to another folder on the same physical hard drive. For the real back up process I will copy from one physical hard drive to another physical hard drive and the speed will be considerably bigger.

If the speed of the backup has a high priority we can buy a SSD hard drive, else if the cost of the backup has a higher priority we can buy a classic hard drive and still achieve a good backup speed.

Image 11 - Copying speed



Performance Chart

Key	Object	Measurement	Rollup	Units	Latest	Maximum	Minimum	Average
	WD-HDD-2-T	Write rate	Average	KBps	52850	56079	3	4497.52
	WD-HDD-2-T	Write latency	Average	Millisecond	12	21	0	2.078
	AdataSSD	Write latency	Average	Millisecond	3	13	0	5.3
	AdataSSD	Write rate	Average	KBps	15	218	0	6.061
	WD-HDD-2-T	Read rate	Average	KBps	26329	27917	0	2199.80

Target	Status	Details	Initiated by
	2d9b4/TEST_April_27_2015_back_up_XP/backup-worker-000001.vmdk		
	12%		root
	Completed		root

8. Restoring a copy of the virtual machine.

I will consider as a base point for a future comparison the size of the C drive from the "backup-worker-win6" virtual machine. The easiest way to restore a copy of the virtual machine that has been backed up is to go to: File → New → Virtual Machine → **Custom** , go through all steps and when it comes to specify a hard disk Select a Disk → **Use an existing virtual disk** and select "backup-worker.vmdk", the virtual disk that we just copied.

Image 12 - Initial C drive disk size

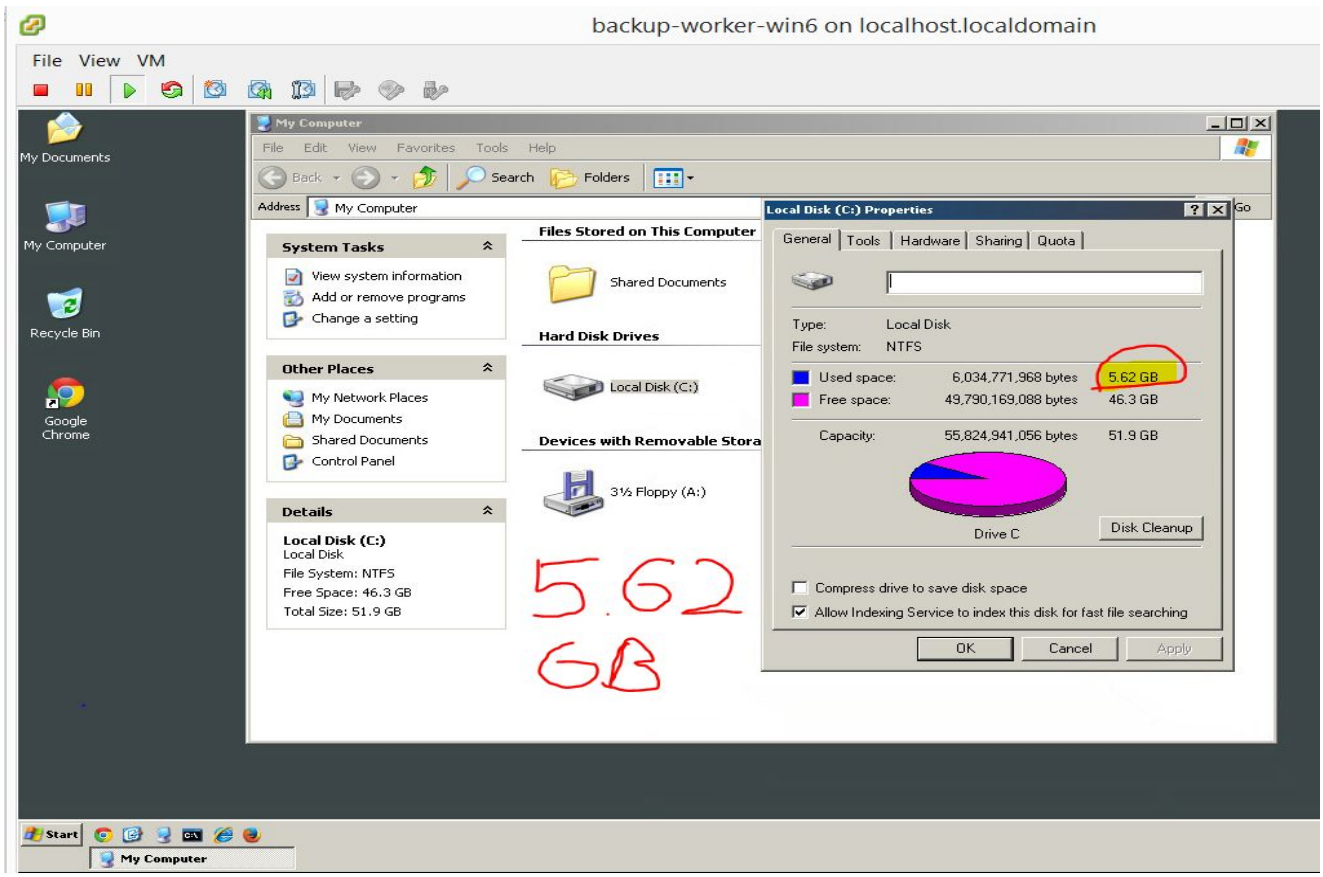
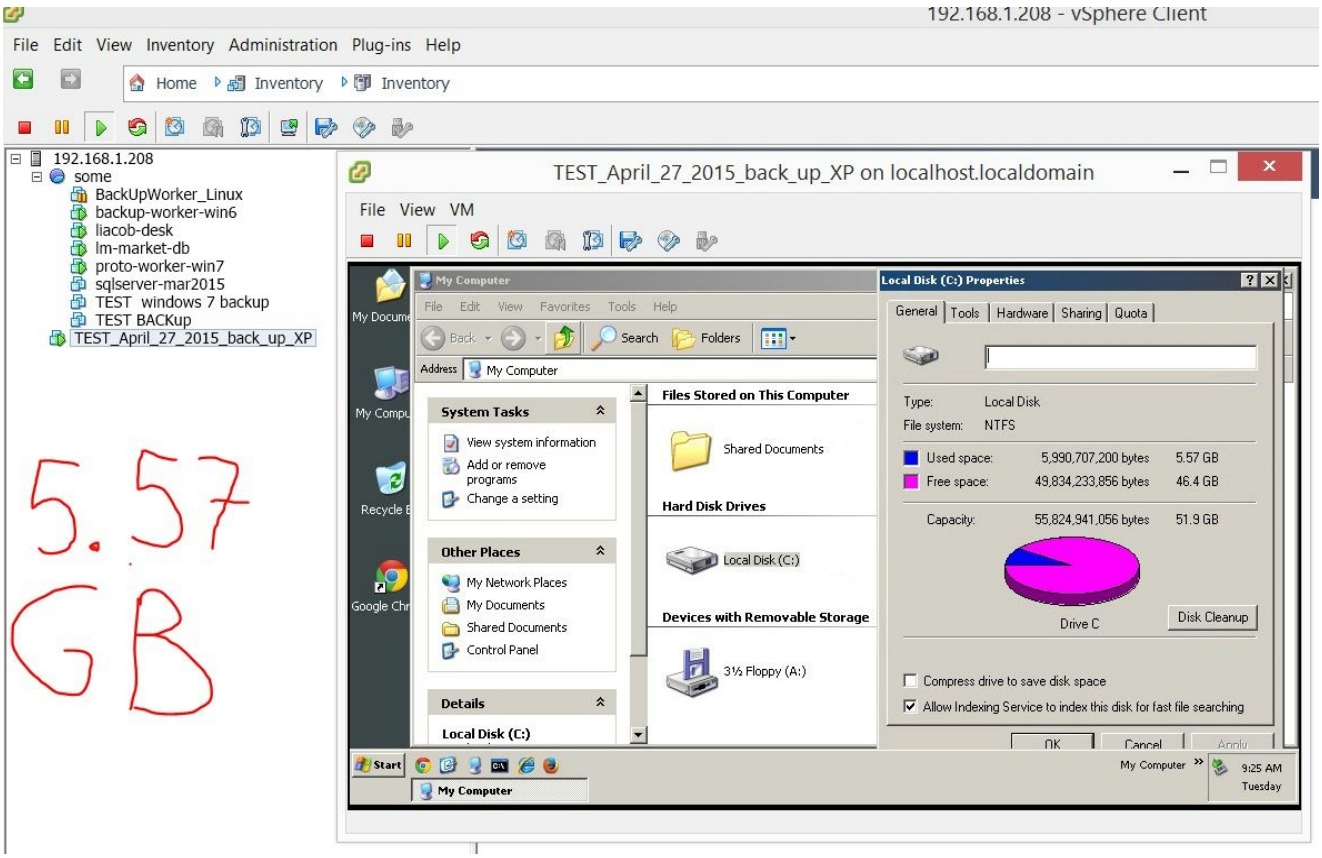


Image 13 - After virtual machine was restored



As we can notice from the two screenshots that I took, the difference between the size of the original C drive and the backed up virtual machine is only of 50 MB. All the programs installed are working perfectly.

This means the backup was **successful.**

9. Find a way to automate the backup process.

After I found a way to make a backup manually now I am thinking how to create a "robot" to do all the actions that I did by himself. After a Google search I found out some names of programs and scripting languages that are promising to record an user actions and then you can run them just by running a script:

- AutoIT v3 - open source
- AutoHotkey - open source
- Action(s) - open source, currently out of support
- FastKeys.

There a lot of couple of articles like *"Automate Just About Anything on Your Windows PC, No Coding Required",* suggesting that they have found some type of software that could automate user defined processes. Of Course none worked out for me and I could not record my actions by clicking and finding all the windows in vSphere.

In some videos on youtube some guys managed to automate some simple tasks, but **coding** in AutoIT or AutoHotkey. So I started learning AutoHotkey as it seemed a better choice from my point of view…
I did managed to lunch vSphere, type in the username and password and login to the ESXi server. It was pretty simple to launch a program and send some keystrokes.

```
Run, VpxClient.exe, C:\Program Files
(x86)\VMware\Infrastructure\Virtual Infrastructure Client\Launcher
Sleep, 1000
Send, root{Tab}passw0rd{Enter}
WinWaitActive, 192.168.1.208 - vSphere Client
Sleep, 2000
Send, {Right 4}
Sleep, 1000
Send, {Shift Down}{F10}{Shift Up}{Down 3}{Right}{Enter}
```

What I am doing in this script is launching vSphere, log in to ESXi server and I'm using the keyboard to navigate to the virtual machine that I'm trying to back up.

This solution was NOT accepted by my supervisor because there was no way to check if my program is taking a snapshot of the targeted virtual machine or just a random one running on the server. Then he suggested to use a programming language that was well developed, open source, general purpose, namely **"Python".**

One quote from the "Zen of Python" says:

"In the face of ambiguity, refuse the temptation to guess.", then I understood what was wrong with my program.

Before I can start writing a program in Python I had to take about 1 month to learn more about programming. A good resource is http://www.codecademy.com/ . Here I could found a good combination between theory and practical exercises.

My supervisor guided me to start learning more about "Python GUI Automation."

After a Google search I found out about a module for Python named "pywinauto". Also this module has a helping tool named "Swapy". With this tool I can inspect Windows applications and get more informations like the window's title and ID. While developing the program I realized that I need a more advanced windows inspection tool, so I found out about "Inspect.exe" part of the Windows SDK for Windows 8.1.

The following program written in Python can launch vSphere, navigate to the target virtual machine, capture it's name, verify if it is the right virtual machine and take a snapshot. (what follows "#"are comments; the use of comments is a good habit for programmers to comment out the code in order to make it more readable and easier to understand for himself and other persons).

```
"""
requires:
  SendKeys  ~ pip install SendKeys
  pywinauto ~ pip install pywinauto
  pyperclip ~ pip install pyperclip
"""
import threading, pyautogui, time, locale, datetime, smtplib,
pyperclip,re
from pywinauto import application
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart

#Set Local time and time format like "Mon May 04 13:09:47 2015"
locale.setlocale(locale.LC_ALL, '')
localtime = time.asctime(time.localtime(time.time()))
# ---------------Launch vSphere application ----------------------
app = application.Application()
app.start_(r"C:\Program Files (x86)\VMware\Infrastructure\Virtual
Infrastructure Client\Launcher\VpxClient.exe")
# Set focus on vSphere and type in the login credentials
app.window_(title='VMware vSphere Client')
app.window_().TypeKeys('root{TAB}popa_jn9A{ENTER}')
time.sleep(2)
app.window_().TypeKeys('{ENTER}')
# ------------- END of connection successfully -------------------

#------------- Wait for  192.168.1.208 - vSphere Client ----------
```

```
window = None
iTries = 0
while True:
  iTries += 1
  try:
     # catch the "Window not found" error and try again
     window = app.window_(title=u'192.168.1.208 - vSphere Client')
     window["TreeView"].Click()
     break
  except Exception as e:
     # if login takes longer than 27 seconds or fails notify the
admin
     if iTries >9:
        print("vShpere faild loging")
        S = smtplib.SMTP('taycor-com.mail.protection.outlook.com',
25)
        Msg = MIMEText("vShpere faild loging", 'Admin attention
required')
        Msg['Subject'] = "BackUpBot"
        Msg['From'] = "butler@taycor.com"
        Msg['To'] = "radupopa2010@yahoo.com"
        S.sendmail(Msg['From'], Msg['To'].split(';'),
Msg.as_string())
        S.quit()
        exit()
     print("Window '192.168.1.42 - vSphere Client' is loading,
please wait ", e)
     time.sleep(3)  #sleep for 3 seconds
  #
#---------- END of Wait for  192.168.1.208 - vSphere Client ------


#------------- After login, send right arrow key 4 times, in order
to navigate to the first virtual machine ----------------------
```

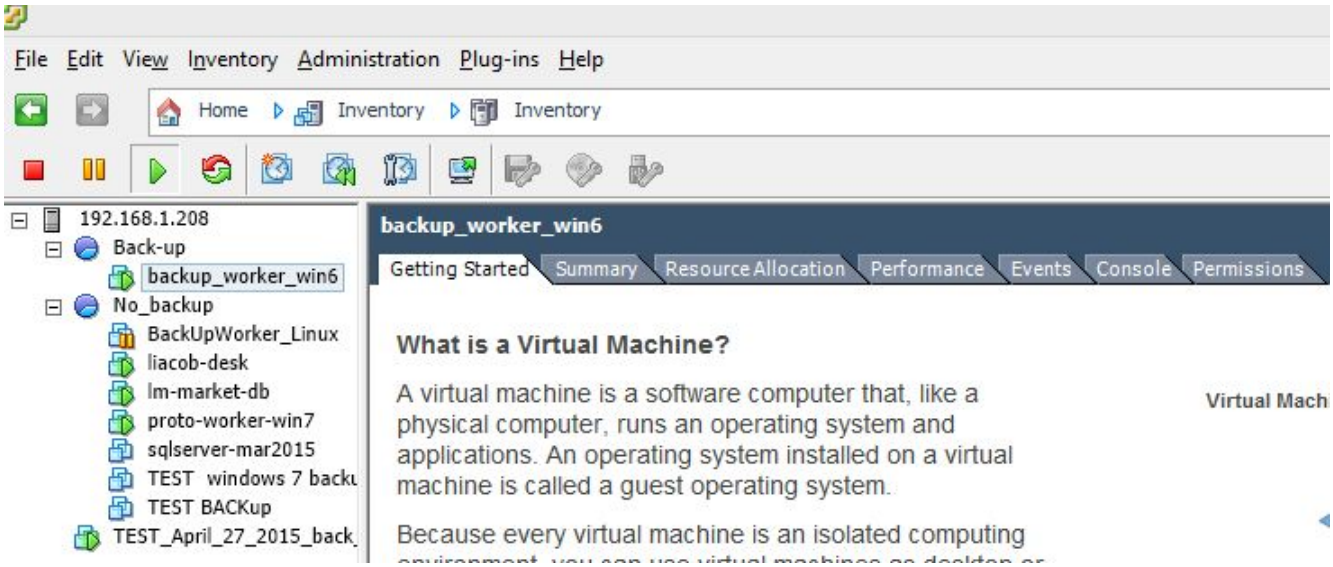
```
time.sleep(3)
window["TreeView"].Click()
window["TreeView"].TypeKeys('{RIGHT}{RIGHT}{RIGHT}{RIGHT}')
time.sleep(2)
# -------------End of "navigate to target VM" --------------------

def ProcessVmSnapshot(app, window, thisVmName):
  # right click on the virtual machine
  # select "Snapshot" from the menu -> select "Tke Snapshot"
  # name the snapshot and paste in the date and time when the
snapshot was taken
  # click "OK"
  import re, time
  window.TypeKeys('+{F10}')
  time.sleep(2)
  window.TypeKeys('{DOWN}')
  time.sleep(1)
  window.TypeKeys('{DOWN}')
  time.sleep(1)
  window.TypeKeys('{DOWN}')
  time.sleep(1)
  window.TypeKeys('{RIGHT}')
  time.sleep(1)
  window.TypeKeys('{ENTER}')
  time.sleep(1)
  snap_window = app.window_(title=u'Take Virtual Machine
Snapshot')
  snap_window.TypeKeys(re.sub(r' ', '{SPACE}', '%s - %s' %
(thisVmName, time.ctime())))
  time.sleep(1)
  # deselect "Snapshot the virtual machine's memory "
  snap_window.TypeKeys('{TAB}')
  time.sleep(1)
```

```
  snap_window.TypeKeys('{TAB}')
  time.sleep(1)
  snap_window.TypeKeys('{SPACE}')
  time.sleep(1)
  snap_window['OK'].Click()
  time.sleep(1)

def GetSelectedVmName():
  #   Send rename command (F2) so I can get access to the VM's
name
  #   copy the auto-selected text (^c) from the clipboard
  #   and then stop the rename by pressing "Escape" from the
keyboard(ESC)
  window.TypeKeys('{F2}')
  time.sleep(1)
  window.TypeKeys('^c{ESC}')
  vmName = str(pyperclip.paste())
  # Use the data in the clipboard to determine the active VM
  return vmName

#-------- Check the if selected Virtual Machine's name is the one
I want to back up ------------------------------
#   If yes, take a snapshot and wait 5 minutes to be sure the
process ended successfully
thisVmName = GetSelectedVmName()
print(thisVmName)
time.sleep(1)  # REMINDER  modify this value when done testing the
program

if thisVmName == "backup_worker_win6":
  ProcessVmSnapshot(app, window, thisVmName)
  time.sleep(1) #sleep for 300 seconds/ 5 minutes
else:
```

```
  # notify the admin, by sending an e-mail
  S = smtplib.SMTP('taycor-com.mail.protection.outlook.com', 25)
  Msg = MIMEText("snapshot failed", 'Admin attention required')
  Msg['Subject'] = "BackUpBot"
  Msg['From'] = "butler@taycor.com"
  Msg['To'] = "radupopa2010@yahoo.com"
  S.sendmail(Msg['From'], Msg['To'].split(';'), Msg.as_string())
  S.quit()
  exit()
#------- Make sure 192.168.1.208 is slected in the left side -----
window.TypeKeys('{LEFT}')
time.sleep(1)
window.TypeKeys('{LEFT}')
time.sleep(1)
window.TypeKeys('{LEFT}')
time.sleep(1)
window.TypeKeys('{LEFT}')
time.sleep(1)
window.TypeKeys('{LEFT}')
time.sleep(1)
window.TypeKeys('{LEFT}')
time.sleep(1)
window.TypeKeys('{LEFT}')
time.sleep(1)
#--------- END of 192.168.1.208 slection process -----------------
```

To be sure that the virtual machine that I want to back up is the first one, so I can access it by sending “right arrow” 4 times from the keyboard, I created a new resource pool and named it “Back-up” (check the snapshot below)

Image 14 - Target virtual machine



After automating the snapshot process, the second phase is to automate the process of copying the the virtual hard disk from the folder containing the all the files from the target virtual machine to a second folder. The following code is able to search for the hard drive containing the folders, search and locate the "backup-worker.vmdk" file, search for the back-up folder and paste it in.

I noticed that when sending "F2" key to rename a file, more information is copied beside the file's name: the file's size is copied as well. This is an example:

```
backup-worker.vmdk

425,984.00 KB

54,525,950.00 KB

Virtual Disk

[WD-HDD-2-T] TEST_April_27_2015_back_up_XP_1

5/12/2015 6:14:13 AM
```

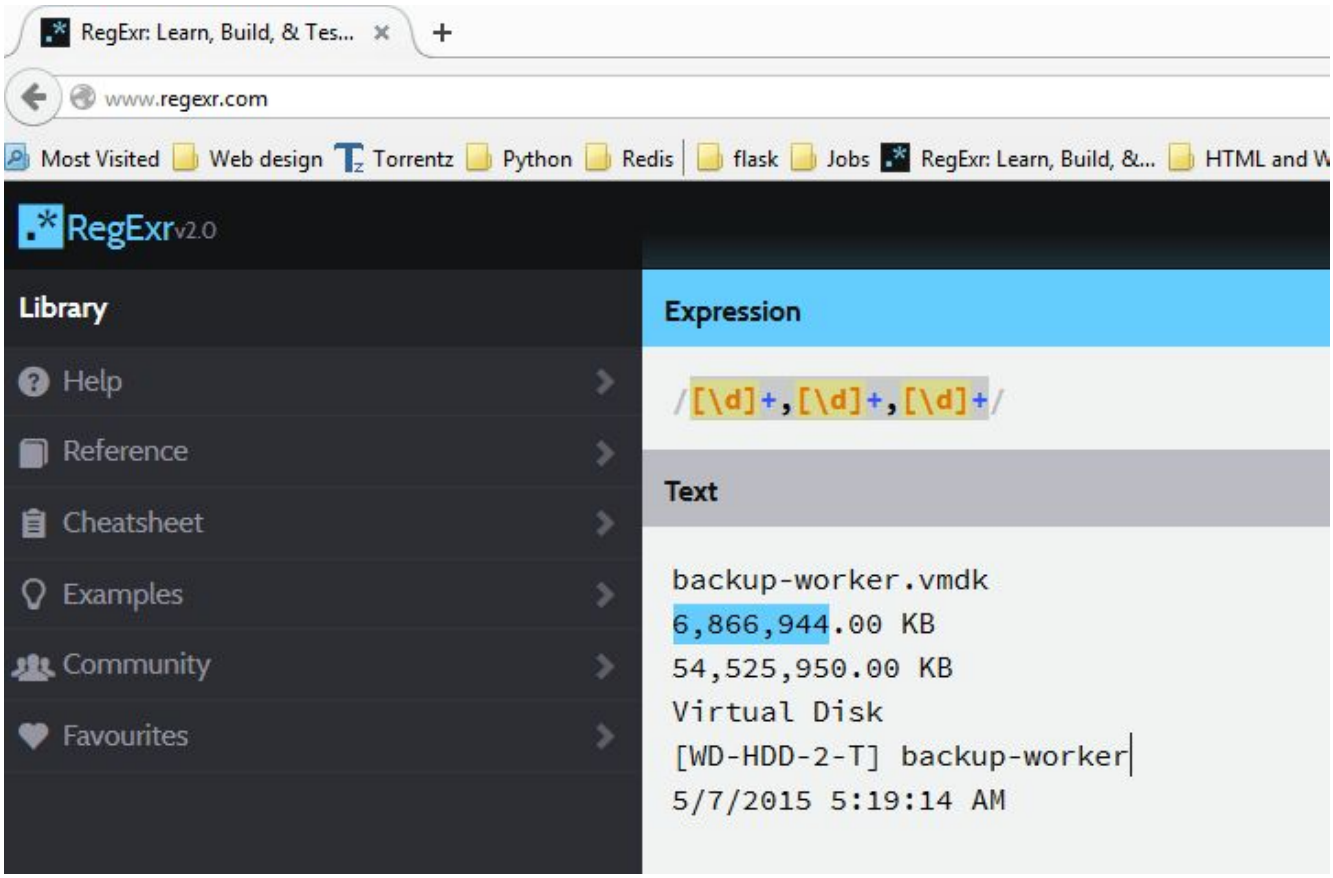
The file size is `425,984.00 KB` . All this information is available as text inside Python. In order to access just the file size, I had to learn a new skill named:

"REGULAR EXPRESSION".

In theoretical computer science and formal language theory, a regular expression (abbreviated regex or regexp and sometimes called a rational expression) is a sequence of characters that forms a search pattern, mainly for use in pattern matching with strings, or string matching, i.e. "find and replace"-like operations. WIKIPEDIA.

With the help of this website : **http://www.regexr.com/** , I could build a regular exression that matches exactly the file size in the info text. See the picture below:

Image 15 - Buliding regular expression pattern



Python has a module called : "re" that uses the power of Regular Expressions to parse text objects.

```
'''
---------------- PART 2 OF THE PROGRAM -------------------------

                COPYING THE VIRTUAL DISK
-----------------------------------------------------------------
'''

import time,re, pyperclip
from pywinauto import application
app = application.Application()
# Make sure I connect to the vSphere application
app.connect_(title=u'192.168.1.208 - vSphere Client')
#'192.168.1.208 - vSphere Client'
```

```
window = app.window_(title=u'192.168.1.208 - vSphere Client')
# Click on the Summary tab
window[u'Summary'].Click()
time.sleep(2)

# Locate the hard drive where
window["ListViewItem-0"].Click()
time.sleep(2)
window["ListViewItem-0"].TypeKeys('{DOWN}{DOWN}')

def checkHardName():
  #   Send right click ( SHIFT+F10 from the keyboard)  I can get access to Rename option and hardware's name
  #   Select everything with CTRL+A, copy the selected text CTRL+A (^c) from the clipboard
  #   and then stop the rename by pressing "Escape" from the keyboard(ESC)
  window.TypeKeys('+{F10}{DOWN}{DOWN}{ENTER}')
  time.sleep(1)
  window.TypeKeys('^c')
  time.sleep(1)
  window.TypeKeys('{ESC}')
  hardName = str(pyperclip.paste())
  return hardName

# Check hdd name
hddNAme = checkHardName()
print(hddNAme)
if hddNAme == "WD-HDD-2-T":
  # Send Right Click and select Brows data store
  window.TypeKeys('+{F10}{DOWN}{ENTER}')
  # wait 2 second for the window to load
  time.sleep(2)
```

```
else:
  # make sure that Western Digital hard is selected
  window["ListViewItem-0"].TypeKeys('{UP}{UP}{DOWN}')
  time.sleep(1)
  hddNAme = checkHardName()
  print(hddNAme)
  if hddNAme == "WD-HDD-2-T":
     # make sure that Western Digital hard is selected
     window.TypeKeys('+{F10}{DOWN}{ENTER}')
     time.sleep(2)
  # wait 2 second for the window to load
# -----------------
# Select the list with the folders containing the virtual machines
window = app.window_(title='Datastore Browser - [WD-HDD-2-T]')
window.Click()
time.sleep(1)
window['tree item'].Click()
time.sleep(1)
window['ListViewSubItem-0'].TypeKeys('{UP}')
time.sleep(1)

# Search for the folder containing the target virtual machine

def checkFolderName():
  window['ListViewSubItem-0'].TypeKeys('{F2}')
  time.sleep(1)
  window['ListViewSubItem-0'].TypeKeys('^c')
  time.sleep(1)
  window['ListViewSubItem-0'].TypeKeys('{ESC}')
  thisFolder = str(pyperclip.paste())
  return thisFolder
```

```
thisVmFolderName = checkFolderName()
print(thisVmFolderName)

# search for the folder backup-worker
while "backup-worker" not in thisVmFolderName:
  window['ListViewSubItem-0'].TypeKeys('{DOWN}')
  thisVmFolderName = checkFolderName()
  print(thisVmFolderName)

# if backup-worker folder is selected press ENTER
if "backup-worker"  in thisVmFolderName:
  window['ListViewSubItem-0'].TypeKeys('{ENTER}')

# search for backup-worker.vmdk file
thisFile = checkFolderName()
while "backup-worker.vmdk" not in thisFile:
  window['ListViewSubItem-0'].TypeKeys('{DOWN}')
  thisFile = checkFolderName()
  print(thisFile)

# if backup-worker.vmdk file was found copy it to the clapboard
if "backup-worker.vmdk"  in thisFile:
  window['ListViewSubItem-0'].TypeKeys('+{F10}')
  time.sleep(1)
  window['ListViewSubItem-0'].TypeKeys('{DOWN}')
  time.sleep(1)
  window['ListViewSubItem-0'].TypeKeys('{DOWN}')
  time.sleep(1)
  window['ListViewSubItem-0'].TypeKeys('{DOWN}')
  time.sleep(1)
  window['ListViewSubItem-0'].TypeKeys('{DOWN}')
  time.sleep(1)
```

```
  window['ListViewSubItem-0'].TypeKeys('{ENTER}')
  time.sleep(1)

# store the file size on backup-worker.vmdk for future reference
fileSize = re.compile(r'[\d]+,[\d]+,[\d]+', re.DOTALL)
mo = fileSize.search(thisFile)
vmdkSize = mo.group()
print('vmdk file size is  ' + vmdkSize)
#--------------------------------

# search for the folder named TEST_April_27_2015_back_up_XP_1
#   first Select the list with the folders congaing the virtual
machines
window = app.window_(title='Datastore Browser - [WD-HDD-2-T]')
window.Click()
time.sleep(1)
window['tree item'].Click()
time.sleep(1)
window['ListViewSubItem-0'].TypeKeys('{DOWN}')
time.sleep(1)
#       second, start the search
thisFile = checkFolderName()
while "TEST_April_27_2015_back_up_XP_1" not in thisFile:
  window['ListViewSubItem-0'].TypeKeys('{DOWN}')
  thisFile = checkFolderName()
  #save all the window info, including file size for later usage
  print(thisFile)

# After finding the folder press ENTER to access it.
if "TEST_April_27_2015_back_up_XP_1" in thisFile:
  window['ListViewSubItem-0'].TypeKeys('{ENTER}')
```

```
# Paste in the "backup-worker.vmdk" file
#window['ListViewSubItem-0'].TypeKeys('{DOWN}')
time.sleep(2)
window['ListViewSubItem-0'].TypeKeys('+{F10}')
time.sleep(1)
window['ListViewSubItem-0'].TypeKeys('{DOWN}')
time.sleep(1)
window['ListViewSubItem-0'].TypeKeys('{DOWN}')
time.sleep(1)
window['ListViewSubItem-0'].TypeKeys('{DOWN}')
time.sleep(1)
window['ListViewSubItem-0'].TypeKeys('{DOWN}')
time.sleep(1)
window['ListViewSubItem-0'].TypeKeys('{DOWN}')
time.sleep(1)
window['ListViewSubItem-0'].TypeKeys('{DOWN}')
time.sleep(1)
window['ListViewSubItem-0'].TypeKeys('{ENTER}')
# sleep 5 seconds so the file starts copying
time.sleep(5)
# -------------- END of COPY - PASTE phase ----------------------
```

Now it would be nice if the “backup robot” could check if the file was copied successfully and send a confirmation e-mail to the administrator. A good strategy to make sure that the virtual disk file was copied successfully is to read the “backup-worker.vmdk” file size from “backup-worker” and store it in a variable in memory; while to file is copying compare at a time interval the file size until the 2 file sizes match.

```
#---------- Verify if the virtual disk file was copied ----------
import time, pyperclip
from pywinauto import application
app = application.Application()
app.connect_(title=u'192.168.1.208 - vSphere Client')
#'192.168.1.208 - vSphere Client'
```

```
window = app.window_(title='Datastore Browser - [WD-HDD-2-T]')

def checkFolderName():
  window['ListViewSubItem-0'].TypeKeys('{F2}')
  time.sleep(1)
  window['ListViewSubItem-0'].TypeKeys('^c')
  time.sleep(1)
  window['ListViewSubItem-0'].TypeKeys('{ESC}')
  thisFolder = str(pyperclip.paste())
  return thisFolder
# ---------------------------

# press refresh button so the new vmdk file is visible
time.sleep(5)
window["refreshButton"].Click()

# Search in the folder TEST_April_27_2015_back_up_XP_1, for the
backup-worker.vmdk file and send call checkFolderName function
filesToCheck = 0    # number of files to check in the folder
thisFile = checkFolderName()
while("backup-worker.vmdk" not in thisFile) and (filesToCheck < 20
):
  window['ListViewSubItem-0'].TypeKeys('{DOWN}')
  thisFile = checkFolderName()
  print(thisFile)
  filesToCheck = filesToCheck + 1

if filesToCheck ==20:
     window['ListViewSubItem-0'].TypeKeys('{UP}{UP}{UP}{UP}{UP}{UP
     }{UP}{UP}{UP}{UP}{UP}{UP}{UP}{UP}')
     time.sleep(3)

filesToCheck = 0
```

```
thisFile = checkFolderName()
while("backup-worker.vmdk" not in thisFile) and (filesToCheck < 20) :
  window['ListViewSubItem-0'].TypeKeys('{DOWN}')
  thisFile = checkFolderName()
  print(thisFile)
  filesToCheck = filesToCheck + 1


# After finding sleep 5 seconds, send checkFolderName again to get file size of vmdk file as a text
filesToCheck =0   # number of files to check
timer = 0     # time  the while loop last and check if
while (vmdkSize not in thisFile) and (timer < 600):   # for my project and testing speed time , 10 minutes
  thisFile = checkFolderName()
  window['ListViewSubItem-0'].TypeKeys('{DOWN}')
  print("copying please wait")
  timer = timer +1
  filesToCheck = filesToCheck +1
  if filesToCheck > 20:
     window["refreshButton"].Click()
     filesToCheck =0
     time.sleep(5)
print("File copied successfully ")
#------------ END of waiting for files to copy -------------------
#------------Send confirmation email code ------------------------
import smtplib
from email.mime.text import MIMEText
S = smtplib.SMTP('taycor-com.mail.protection.outlook.com', 25)
Msg = MIMEText("Virtual disk copied SUCCESSFULLY", 'Virtual disk copied SUCCESSFULLY')
Msg['Subject'] = "BackUpBot"
Msg['From'] = "butler@taycor.com"
```

```
Msg['To'] = "radupopa2010@yahoo.com"

S.sendmail(Msg['From'], Msg['To'].split(';'), Msg.as_string())

S.quit()

exit()

#-----------End of confirmation email code ----------------------
```

After performing so many tests on one virtual machine I realized that TWO more step have to be done after the virtual disk file was copied:

1. the snapshot has to be deleted; by doing this the virtual machine will start writing data to the "backup-worker-.vmdk" file again instead of the new created file "backup-worker-000001.vmdk".

I will not present the code for this final steps because my paper will become very long;

2. the "older" version of the virtual disk has to be deleted.

Now my program needs to run every Friday evening at 19:00, when nobody works at that hour. The first option and the easiest is to create a weekly task with "Task Scheduler" from "Windows Administrative Tools". When I opened this tool, I noticed that there were already some tasks created, like "GoogleUpdateTaskMachineCore". If this tool is good enough for Google to schedule updates, then it means it good enough for me also.

To create a new weekly task I just have to do the following steps: Create new basic task → Name: "Vm Backup Robot" → Next → Trigger: check weekly → Next → check Friday and set the hour → Next → Action: check Start a program → Next → Brows for the program, in my case I will select "today.py" (the name of my program) → Next → Finish .

This task will be created on a special virtual machine like "worker" that does some other automated task created by the IT department.

10. How to improve the developed program ?

While developing this "GUI automation program", a personal curiosity sprouted in my mind: "How do backup software work in general ? " .
I found out that IT professionals try to conserve the resources by synchronizing files between the production server and the backup server. In Unix like operating systems, the most known open source utility is "RSYNC" . Rsync was initially developed in 1996 and since then it is continuing being used and maintained. The last release is 3.1.1 (June 22, 2014; 10 months ago). It is written in C , which is a low level programming language and has a GNU General Public License.
With this in mind, I started searching if there are any open source software already written for ESXi that uses Rsync. The result :

XSI BACKUP - Free Backup Software for VMware ESXi VMs.
http://sourceforge.net/projects/xsibackup/

Features:

- **Works with ESXi Free version**.

- Command line tool with real time output. Self contained, no dependencies, runs in the hypervisor.

- Hot backups, no downtime.

- Unattended backups of ESXi servers.

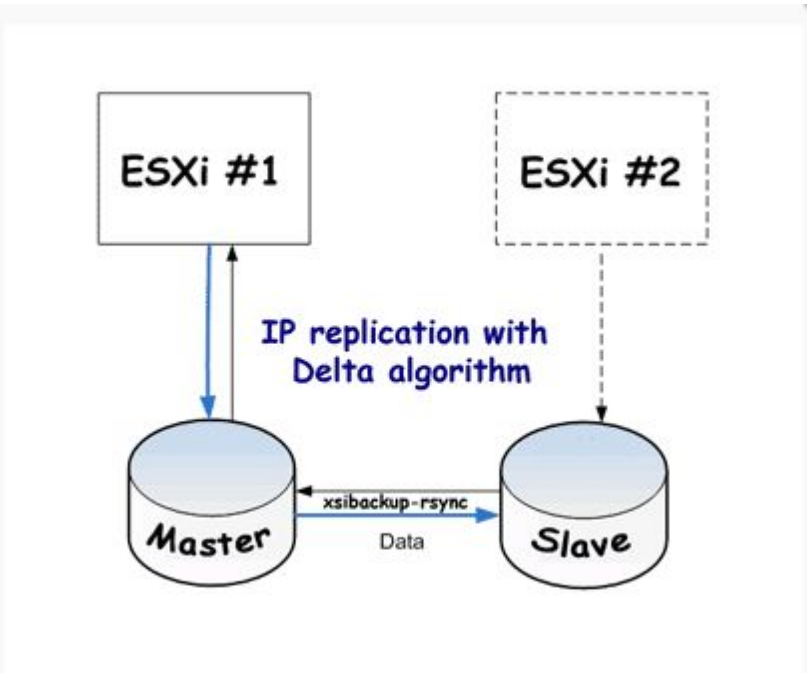
- Automatic provisioning of space by deleting the older folders.

- Detailed report of all the backup process per virtual machine (speed, shutdown, boot, space provisioning, times, etc...)

- Cron programmable.

Analyzing the features : "Works with ESXi Free version" is perfect for my situation, "Hot backups, no downtime" means can backup running virtual machines, no powering down is required, "Cron programmable" - Cron is the default "Task Scheduler" for Linux operating systems . I got a perfect match for my case of study.

Image 16 - XSI Backup software



Installing this software is not as simple as installing a Windows software (next → next → finish). Some skills in Linux operating systems are required. After all the installation and configuration work was done, here is the command that I can issue from the terminal so the backup process starts on my demand:

```
./ xsibackup \
     --time = "Fri 04:22" \
     --backup-point=/vmfs/volumes/backup --backup-type=running \
     --mail-from=butler@taycor.com --mail-to=rbutler@taycor.com \
     --smtp-srv=crmserver --smtp-port=25 --test-mode=true
```

A command line software can span multiple lines by adding a backslash "\" character at the end of the line.

Taking a closer look at XSI Backup software I noticed that is composed of 2 files: "xsibackup-rsync," a compiled binary executable; and "xsibackup," an executable bash script readable text file. By opening "xsibackup" I could say it is a"shell script". In UNIX like operating systems, the way to figure it out in what language a program is written in is to look at the first line `"#!/bin/sh"` . The "#!" is called "shebang" and what follows is the path to the interper that program uses, in our case "/bin/sh" is the Bourne shell or a compatible shell. Next, the "xsibackup-rsync" file is a compiled code version of Rsync and Cron. The two Linux utilities normally can not be installed on ESXi because as I mentioned previously in this paper, ESXi is built on the Linux kernel though it lacks dynamic library Linux distribution, many components of an operating system are missing. By statically compiling the code and the downloading it directly in the ESXi , the machine can "understand" it and run it.

After running a test backup with XSI Backup, this is the out from the confirmation e-mail:

VIRTUAL MACHINE BACKUP AT HOST:

localhost

Available room in device /vmfs/volumes/dsMushkin480-Sept14/xsibackup/arc-primary before backup: 271 Gb.
Needed room in device /vmfs/volumes/dsMushkin480-Sept14/xsibackup/arc-primary for backup: 152 Gb.

VM Name	State	Size	Stop	Copy	Start	Time (min)	Speed (mb/s)
sqlserver-oct2014.1	on	152G	NO (hot backup)	OK	-	12	205

Available space in device /vmfs/volumes/dsMushkin480-Sept14/xsibackup/arc-primary after backup: 183 Gb.

Complete backup elapsed time: 12 min

11. ROBO-Backup VS XSI Backup.

You may be wondering “Who the heck is ROBO-Backup ?” Well, I decided to name my Python program: ROBO-Backup, because each software has a name.

ROBO-Backup features and advantages:

1. Works with ESXi Free version.
2. Running virtual machines backups, no downtime.
3. Time scheduler availability.
4. E-mail report.
5. Custom features can be added later on.
6. Free, no licensing cost.
7. Can be adapted to work with future versions of vShere.
8. Speed of backup, in my case of study was 3MBps.

Disadvantages:

1. Requires to run on a virtual machine, physical resource consuming.
2. Not easy to port to a different operating system, it requires Python and some additional Python modules.

XSI Backup features and advantages:

1. - Works with ESXi Free version.
2. - Command line tool with real time output. Self contained, no dependencies, runs in the hypervisor.
3. - Hot backups, no downtime.
4. - Unattended backups of ESXi servers.
5. - Automatic provisioning of space by deleting the older folders.
6. - Detailed report of all the backup process per virtual machine (speed, shutdown, boot, space provisioning, times, etc...)
7. - Cron programmable.
8. Speed of backup, according to the email report: 200 Mbps.
9. Free, no licensing cost.

Disadvantages:

1. Delta algorithm can be slow on big files, in fact from an efficiency point of view it's only a way to change bandwidth need for CPU cycles.
2. Some people have posed their concerns about installing an alien binary in their ESXi hosts.
3. One day the company may change the licensing type.

As I can see both solutions are pretty close. My supervisor chose to use **XSI Backup** mainly because of the speed of the backup, but he is happy to know he has a *backup alternative to the backup solution :).*

12. Conclusion.

When I started working on this task at the beginning I was thinking that it will be a waste of time trying to find a "workaround" to an intentional limitation implemented by a big software company like VMware.

For me, personally, it turned out to be an inestimable learning experience. I have developed skills I did not had before, like learning to program in Python. Once again it is confirmed the fact that nowadays the information we are searching for is "there" some place on the Internet, but we just don't know the right "keyword" to search for it on Google or some other search engine that doesn't trank our history and personal information like DuckDuckGo(at least this is what they say officially).

For Taycor Financial it worth not spending approximative $5000, the cost of a permanent license with backup features from VMware and 1 month work of an intern. Of Course I couldn't go so far in such a short time without the guidance of my technical supervisor, Robert Butler.

Doing an internship it was a GREAT learning opportunity and I would like to thank the persons responsible for making it a mandatory requirement in our study curriculum.

13. Appendix - links related to this project

https://my.vmware.com/web/vmware/details?downloadGroup=VDPADV55_7&productId=375&rPId=6571#open_source

http://en.wikipedia.org/wiki/BIOS

http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2006202

www.google.com

www.duckduckgo.com

http://33hops.com/blog_xsibackup-rsync-considerations.html

http://sourceforge.net/projects/xsibackup/

https://www.python.org/

www.http://stackoverflow.com/

www.wikipedia.org

https://code.google.com/p/pywinauto/