2013-09

Big data analytics test bed

Doucet, Rachel A.; Dontchev, Deyan M.; Burden, Javon S.; Skoff, Thomas L.

Monterey, California: Naval Postgraduate School

https://hdl.handle.net/10945/37615

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

APPLIED CYBER OPERATIONS CAPSTONE PROJECT REPORT

BIG DATA ANALYTICS TEST BED

by

Rachel A. Doucet
Deyan M. Dontchev
Javon S. Burden
Thomas L. Skoff

September 2013

Thesis Advisor:	Mark Gondree
Co-Advisor:	Thuy Nguyen

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704–0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202–4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704–0188) Washington, DC 20503.			
1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** September 2013	**3. REPORT TYPE AND DATES COVERED** Capstone Report	
4. TITLE AND SUBTITLE BIG DATA ANALYTICS TEST BED			**5. FUNDING NUMBERS**
6. AUTHOR(S) Rachel A. Doucet, Deyan M. Dontchev, Javon S. Burden, Thomas L. Skoff			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943–5000			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number ____N/A____.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			**12b. DISTRIBUTION CODE** A

13. ABSTRACT (maximum 200 words)

The proliferation of big data has significantly expanded the quantity and breadth of information throughout the DoD. The task of processing and analyzing this data has become difficult, if not infeasible, using traditional relational databases. The Navy has a growing priority for information processing, exploitation, and dissemination, which makes use of the vast network of sensors that produce a large amount of big data. This capstone report explores the feasibility of a scalable Tactical Cloud architecture that will harness and utilize the underlying open-source tools for big data analytics.

A virtualized cloud environment was built and analyzed at the Naval Postgraduate School, which offers a test bed, suitable for studying novel variations of these architectures. Further, the technologies directly used to implement the test bed seek to demonstrate a sustainable methodology for rapidly configuring and deploying virtualized machines and provides an environment for performance benchmark and testing. The capstone findings indicate the strategies and best practices to automate the deployment, provisioning and management of big data clusters. The functionality we seek to support is a far more general goal: finding open-source tools that help to deploy and configure large clusters for on-demand big data analytics.

14. SUBJECT TERMS Big Data, Hadoop, Serengeti, Cloud Computing, Virtualization, Virtual Technology, Multi-level Security, VMware, vSphere 5.1, ESXi, Virtualized Hadoop,			**15. NUMBER OF PAGES** 155
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

NSN 7540–01–280–5500

Standard Form 298 (Rev. 2–89)
Prescribed by ANSI Std. 239–18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

BIG DATA ANALYTICS TEST BED

Rachel A. Doucet
Javon S. Burden
Deyan M. Dontchev
Thomas L. Skoff

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN APPLIED CYBER OPERATIONS

from the

NAVAL POSTGRADUATE SCHOOL
September 2013

Author: Rachel A. Doucet
Deyan M. Dontchev
Javon S. Burden
Thomas L. Skoff

Approved by: Mark Gondree
Capstone Project Advisor

Thuy Nguyen
Project Advisor

Cynthia Irvine
Chair, Cyber Academic Group

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The proliferation of big data has significantly expanded the quantity and breadth of information throughout the DoD. The task of processing and analyzing this data has become difficult, if not infeasible, using traditional relational databases. The Navy has a growing priority for information processing, exploitation, and dissemination, which makes use of the vast network of sensors that produce a large amount of big data. This capstone report explores the feasibility of a scalable Tactical Cloud architecture that will harness and utilize the underlying open-source tools for big data analytics.

A virtualized cloud environment was built and analyzed at the Naval Postgraduate School, which offers a test bed, suitable for studying novel variations of these architectures. Further, the technologies directly used to implement the test bed seek to demonstrate a sustainable methodology for rapidly configuring and deploying virtualized machines and provides an environment for performance benchmark and testing. The capstone findings indicate the strategies and best practices to automate the deployment, provisioning and management of big data clusters. The functionality we seek to support is a far more general goal: finding open-source tools that help to deploy and configure large clusters for on-demand big data analytics.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

AD DS	Active Directory Domain Services
C5I	Command, Control, Communications, Computers, Combat Systems and Intelligence
CANES	Consolidated Afloat Networks and Enterprise Services
CDH3	Cloudera’s Distribution Including Apache Hadoop version 3
CDH4	Cloudera’s Distribution Including Apache Hadoop version 4
CISR	Center for Information Systems Security Studies and Research
COS	Console Operating System
DB	Database
DB2	Database Two
DCUI	Direct Console User Interface
DHCP	Dynamic Host Configuration Protocol
DIMM	Dual Inline Memory Module
DISA	Defense Information Systems Agency
DNS	Domain Name System
DoD	Department of Defense
ERN	Education Remote Network
HA	High Availability
HDFS	Hadoop-Distributed File System
IaaS	Infrastructure as a Service
IP	Internet Protocol
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
IT	Information Technology
KVM	Keyboard Video Monitor
LDAP	Lightweight Distributed Access Protocol
LTE	Limited Technology Experiments
MILCOM	Military Communications
NCW	Network Centric Warfare
NIC	Network Interface Card

NIST	National Institute of Standards and Technology
OVA	Open Virtual Appliance
OVF	Open Virtual Format
PaaS	Platform as a Service
SaaS	Software as a Service
SELinux	Security-Enhanced Linux
SP2	Service Pack Two
SQL	Structured Query Language
SSO	Single Sign-On
SSPI	Security Support Provider Interface
STS	Security Token Service
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
UPS	Uninterruptible Power Supply
USB	Universal Serial Bus
vApp	Virtual Appliance
VM	Virtual Machine
VMFS	Virtual Machine File System

I. INTRODUCTION

A. MOTIVATION

1. Cloud Computing

Cloud computing has revolutionized the way ahead for Information Technology (IT). It has changed the physical and logical architecture of the business. It can be described as:

> A large-scale distributed computing paradigm that is driven by economies of scale, in which a pool of abstracted, virtualized, dynamically-scalable, managed computing power, storage, platforms, and services are delivered on demand to external customers over the Internet. [1]

In general, cloud computing is a colloquial expression with varying interpretations; however, it is commonly expressed in terms of anything that involves the delivery of hosted service(s) via the Internet on demand. In broad terms, the hosted services are broken down into three categories: Software as a Service (SaaS), Platform as a Service (PaaS), or Infrastructure as a Service (IaaS). Cloud computing has transformed the IT infrastructure to provide scalability, rapid deployment, full transparency for managing operating costs, elastic services and shared resources. The cloud has a vital role on how we align IT to support mission and business requirements. Various organizations and entities attempt to define "Cloud Computing," creating ambiguity with the "true" definition. While ambiguity does exist with the definition, the common goal behind cloud computing remains the same. In September 2011, the National Institute of Standards and Technology (NIST) defined cloud computing as: "A model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model is composed of five essential characteristics, three service models, and four deployment models" [2]. This has become the most widely accepted definition of cloud computing. NIST lists the same five essential characteristics: On-Demand Self

Service, Broad Network Access, Resource Pooling, Rapid Elasticity, and Measured Service [2].

Cloud computing relies on sharing resources instead of having local physical servers handle specific applications or services. This Internet-based computing moves the work from an organization's resources, such a physical servers, computers, applications, and devices to the Internet. The IT infrastructure is shared between pools of systems linked together. Through virtualization, cloud computing is maximized through on-demand delivery of computing resources to global customers in a cost effective manner.

As the U.S. Navy moves more of its operations to cloud computing models, it will simplify the overall administration and oversight of its IT infrastructure. This allows IT organizations to focus more of their efforts on addressing mission requirements and needs. The elasticity of cloud architectures affords organizations the dynamic deployment needed depending on their specific mission. The most widely used terms are the public and private clouds. Public clouds share resources as a service over an Internet connection, whereas in private clouds, the cloud is honed behind a firewall with internally managed services. The Department of Defense (DoD) is aggressively seeking out cloud adoption due to its scalability, elasticity, mobility, and reduced overhead. Most organizations, whether DoD or civilian, are seeking to reduce the operational costs of their IT resources. The Navy is moving toward an innovative approach of the private cloud as a strategic enabler for accelerating the continuous evolution of communication networks to achieve optimal performance. The Navy Tactical Cloud and Intelligence Cloud are supporting initiatives deployed to meet such net-centric performance.

Cloud computing is being adopted by the Military because it enables convenient, on-demand network access to a shared pool of configurable computing resources (networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. To address the emerging warfighter needs for enhanced Command, Control, Communications, Computers, Combat Systems, and Intelligence (C5I), and other IT capabilities, cloud computing moves the applications, data, and computing from traditional workstations and desktops to a modular, shared computing tactical cloud using virtualization at the Tactical Operations Center.

2. Navy Tactical Cloud

Cloud computing offers a paradigm shift in the way IT services are delivered. When cloud computing is combined with virtualization, the benefits to an IT infrastructure prevail over traditional computing. Currently, Defense Information Systems Agency (DISA) is the DoD's only cloud service provider for all Naval shore facilities. The Navy's Consolidated Afloat Networks and Enterprise Services (CANES) program is transitioning it's afloat IT environments to cloud based computing. The Intelligence communities have also begun to operationalize private cloud architectures that process data with different classifications. In particular, various new and open-source technologies have been adopted in the context of the Naval Tactical Cloud and the Intelligence Cloud. Prominent among these technologies are those which allow for the disseminated processing of large data sets across clusters of nodes and computers, designed to scale up from single servers to thousands of machines. The Naval Tactical Cloud and the underlying architectures support the tactical "big data" analytics that provide shared situation awareness, encompassing all domains among geographically dispersed forces in a digitally connected battlespace.

In the battlespace environment, enormous amounts of data are being collected, stored, and disseminated to combatant commanders, warfighters, and decision makers. As Navy sensors evolve, the volume, variety, velocity, and variability will expand on a daily basis. The Navy demands big data cloud technologies to provide capability and agility and ensure life cycle costs are kept to a minimum.

Tactical computing encompasses "all computations necessary to provide shared situational awareness among geographically dispersed forces in a digitally connected battlespace" [3]. Aligned with Admiral Vern Clark's Sea Power 21 initiative, "The 21st century sets the stage for tremendous increases in naval precision, reach, and connectivity, ushering in a new era of joint operational effectiveness. Innovative concepts and technologies will integrate sea, land, air, space, and cyberspace to a greater extent than ever before. In this unified battlespace, the sea will provide a vast maneuver area from which to project direct and decisive power around the globe" [4]. The Naval Tactical Cloud and Intelligence Cloud introduce innovative capabilities to achieve unprecedented maritime power and enhance decisive superiority in order to dominate the unified battlespace anytime, anywhere.

B. STRUCTURE

Chapter I addresses the motivation behind the virtualized test bed cloud infrastructure. It briefly discusses the necessity for seeking open-source tools which ease the deployment and configuration of large clusters for big data analytics. It illustrates the evolving proliferation of data throughout the DoD and the growing popularity for information processing, dissemination, and storage within the Navy. The Navy Tactical Cloud and the Intelligence Cloud are infrastructures are discussed as a way to simplify management and oversight of the DoD IT infrastructure.

Chapter II incorporates a Literature Review of several topics relative to the virtualized test bed in our cloud computing environment. We focused on three primary areas: VMware vSphere 5.1, Apache Hadoop, and Project Serengeti. The VMware vSphere 5.1 suite encompasses several sub categories, such as the VMware ESXi hypervisor, vCenter Server, vCenter Single Sign-On Server, vCenter Inventory Service, and vCenter Inventory Tagging. The development and increasing popularity of Apache Hadoop is also further described. This section was also divided into two significant components: MapReduce and the Hadoop Distributed File System (HDFS). We included the advantages of these powerful tools when dealing with large data sets. Further detail is given to HDFS and how it operates its cluster in a Master and Slave architecture using Name Nodes and Data Nodes. Lastly, this chapter describes Project Serengeti, a virtual appliance which is employed to automate deployment and management of Apache Hadoop clusters on VMware vSphere platforms. The architecture and the seven step process of deploying a Hadoop cluster is described in this section.

Chapter III displays our Project Overview, which is divided into four phases (Phase I-Phase IV). Each phase, summary, status, and appendices is illustrated in Table 1. Each phase is further defined in succeeding chapters.

Chapter IV describes Phase I in further detail, which outlines the hardware and software of the Center for Information Systems Security Studies and Research (CISR) big data test bed. It is divided into sections, such as the hardware upgrade, vSphere ESXI 5.1 upgrade, the installation of vCenter Server 5.1, and vSphere Client 5.1. Other required services, such as active directory support and database support were also discussed in this Chapter. A visual representation is provided in Figure 9 for hardware and software component details and their network configurations.

Chapter V describes Phase II, which uses Serengeti to install Hadoop cluster with its default operating system, CentOS 5.6. This chapter incorporates the installation of Hadoop and the deploying of clusters using Project Serengeti. It also defines the configuration and customization parameters within Serengeti. The architecture overview, encompassing sequential provisioning steps to reduce deployment time is also illustrated in this chapter. The software, network, and resource requirements are also introduced which support Serengeti's virtual appliance. The subsequent sections in Chapter V provide information on the Serengeti virtual appliance installation.

Chapter VI describes Phase III of the project and explores our first attempt to modify the Serengeti virtual appliance (vApp) to use Fedora 18 vice the default CentOS 5.6 operating system. In addition to modifying the template, we attempt to deploy and provision Hadoop clusters. This chapter illustrates the major challenges we have faced with Fedora 18 and our observations throughout the phase in addition to our successes and failures.

Chapter VII describes our final phase, Phase IV, which explores the automation process of Hadoop clusters by cloning with a modified template virtual machine (VM) with a Fedora 13 operating system. This operating system used in the CISR MLS-aware Hadoop test bed. In light of the wider range of available resources for Fedora 13, documented experience with Serengeti was extremely scarce. This chapter illustrates the major challenges we faced and the resources and strategies we used to successfully complete this phase.

Chapter VIII summarizes the project, outlining our accomplishments, lessons learned, and remaining work for future research.

II. LITERATURE REVIEW

A. VMWARE VSPHERE 5.1

Headquartered in Palo Alto, California, VMware is a cloud computing and virtualization software provider with a wide portfolio of products and services. The company's core concentrations are cloud-computing services, administrative and collaboration tools, and software applications. This review will focus on VMware's vSphere 5.1 datacenter virtualization and management platform and the components that are essential for administering the datacenter. vSphere is the virtualization enterprise suite for VMware's cloud computing virtual infrastructure. Together, the functionality of these software and hardware components can be thought of as a "cloud operating system." VMware's vSphere 5.1, released in August 2012, encapsulates two core components, (1) VMware ESXi hypervisor and (2) VMware vCenter Server. Next, we review the software stack comprising vSphere 5.1.

1. VMware ESXi

At the core of vSphere's virtual architecture is the ESXi server. The ESXi software is a *hypervisor*, the main software that manages and controls the virtualization layer on a physical server (see Figure 1). VMware's ESXi hypervisor is radically distinctive from the company's classic ESX 3.x and 4.x hypervisors, which it superseded. In ESXi, VMware removed the (Linux OS based) *vmnix* service console, which performed all of the local management tasks such as executing scripts and installing third-party agents for hardware monitoring, backup or systems management. Currently, management functionality has migrated to remote management tools. This new compact architecture (less than 150MB vs. 2GB) is designed for integration directly into virtualization-optimized server hardware, enabling rapid installation, configuration, and deployment [5]. Leading server manufacturers such as Dell, HP, Fujitsu, IBM, and Siemens are now building the VMware hypervisor directly into their x86 servers. As a layer that operates independently from any general-purpose operating system, ESXi claims to offer improved security, increased reliability, and a simplified management console.

The ESXi hypervisor only runs on specific hardware platforms and support for unnecessary devices has been removed, thus vastly reducing the kernel code [6]. With the removal of the *vmnix* service console, all agents now run directly on the *vmkernel* and management functionality is pushed to remote management tools. The *vmkernel* manages the guest's access to the host's physical hardware, providing CPU scheduling, memory management, and virtual switch data processing. All infrastructure services are provided natively through modules included with the *vmkernel*. Other authorized third party modules, such as hardware drivers and hardware monitoring components, can run in *vmkernel* as well. For security considerations, only digitally-signed VMware modules are permitted on the system, minimizing the introduction of arbitrary code [7]. Figure 1 provides a simplified overview of the vSphere ESXi architecture.

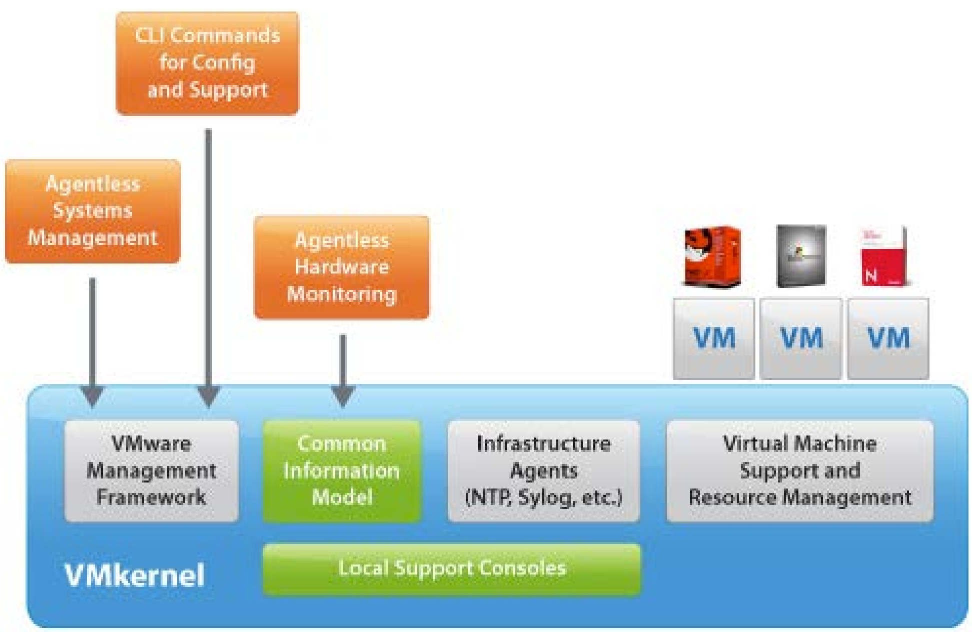


Figure 1. VMware ESXi Architecture (from [7]).

2. vCenter Server

vCenter Server is a centralized management utility for ESXi hosts and their respective virtual machines deployed within the vSphere infrastructure (see Figure 2). Essentially, it acts as a management proxy that executes all administrative functions on ESXi hosts. Unlike ESXi, vCenter Server is licensed and sold separately and runs on a dedicated Windows Server (or Window's VM). From a single console, network administrators have visibility into every level of the virtual infrastructure. In the absence of vCenter Server, network/system administrators would face a number of challenges such as independently managing all ESXi hosts, inability to create clusters and share resources, and the inability to migrate VMs between hosts. Through vCenter Server, the deployment, management, automation, and security services are centralized from a single console. To enhance scalability, vCenter Server depends on a backend database (Microsoft SQL Server, Oracle, or IBM DB2) to store data about the managed hosts and VMs [6]. With the appropriate licensing scheme, vCenter extends the capabilities of the hosts it manages.

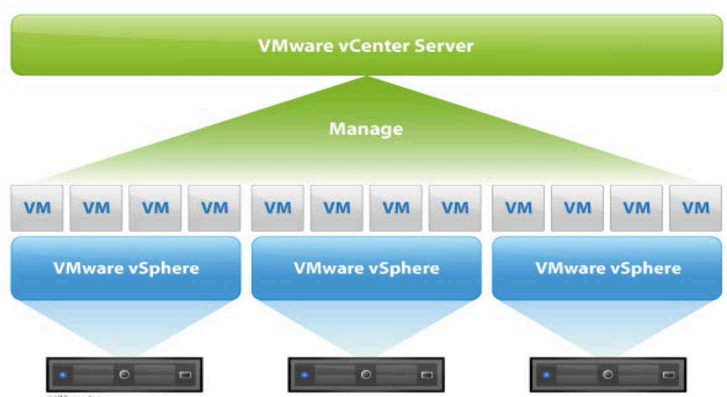


Figure 2. VMware vCenter Server Architecture (from [8]).

VMware's vSphere 5.1 introduced a number of new features supported by vCenter. The three most notable components include: *vCenter Single Sign-On Server*, *vCenter Inventory Service*, and *vCenter Inventory Tagging*.

a. vCenter Single Sign-On

In vSphere 5.1, the Single Sign-On (SSO) service is a crucial component of the vCenter Server suite. The SSO component centralizes authentication service used by the vCenter Server, enabling vSphere software components and authorized users to authenticate through a secure token exchange. The SSO integrates with Active Directory and lightweight directory access protocol (LDAP) services for authentication. When users log into vCenter, a token is issued to the SSO database, which authenticates the user(s) against the configured identity source (Active Directory or OpenLDAP). Once authenticated, the username and password gets substituted for a security token, which in turn is used to access the desired vCenter component(s). Figure 3 summarizes the SSO authentication process.

The Single Sign-On component must be installed before any portion of vCenter 5.1 is installed. During the SSO installation, the following components are also deployed: Security Token Service (STS), Administrative Server, vCenter Lookup Service, and the RSA Security Support Provider Interface (SSPI) service.

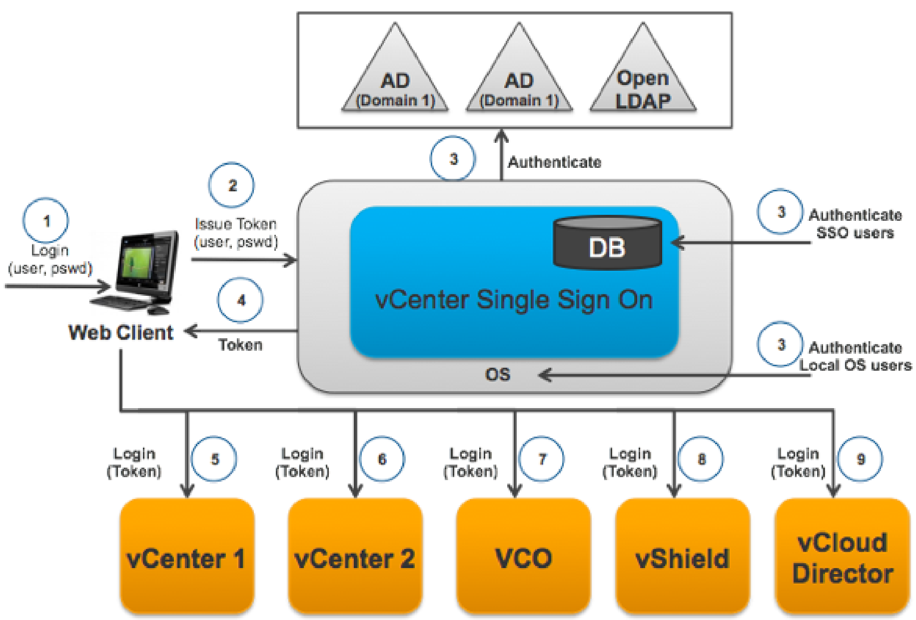


Figure 3. Single Sign-On Authentication Process (from [9]).

b. vCenter Inventory Service

The vCenter Inventory Service minimizes the processor load on the vCenter Server by caching connections, queries, and client requests. The Inventory Service's primary role is to manage the vCenter Web Client inventory objects and property queries requested by clients when users navigate the vCenter environment. Installed as an independent component, the vCenter Inventory Service supports the discovery and management of objects within the vCenter architecture.

c. vCenter Inventory Tagging

The Inventory Tagging service optimizes the client-server communication channels by enabling users to create and add inventory object-level tags. These tags are then used to organize and provide faster retrieval with inventory queries [10].

B. APACHE HADOOP

1. Hadoop Development

The amount of digital data being generated and stored has grown exponentially in recent years. Data once measured in gigabytes, is now measured in terabytes, petabytes and exabytes. Conventional database systems are not able to keep up with the demands of massive data aggregation. The way we handle data has evolved due to these demands. The Hadoop filesystem solution was created to help process data, leveraging clusters of relatively low-cost servers. Costs grow linearly with the number of servers, and there is no ultimate limit, in comparison to relational databases.

The processing thresholds of traditional database systems are incompatible with the massive data processing requirements that companies such as Google, Yahoo, and Facebook require for their data. They require advanced tools to search and process large amounts of data efficiently. For some organizations, the size of these datasets is directly attributable to significant global trends, such as the social media explosion, rise of global ecommerce, popularity of smart mobile devices, and the data collection from sensors and ubiquitous computing devices. For these organizations, the ability to conventionally consolidate, search and analyze datasets is overwhelmed.

Globally, organizations are racing to develop and deploy big data analytic methodologies in order to take advantage of the obscured opportunities and insights within their datasets. As big data analytics become a necessity, relational databases struggle with variety of data input, such as structured, unstructured, semi-structured and complex data. These issues motivate the MapReduce programing model and led to the Apache Hadoop project, which presents a framework for distributed analytical processing over big data.

2. Hadoop Structure

Apache Hadoop provides a suite of open-source software tools for distributed computing. It is a software library that allows for the distributed processing of large data sets across clusters of computers. Its features include the ability to scale up from single to multiple machines while handling failures at the application layer. Hadoop is comprised of several core modules: *Hadoop common*, *HDFS*, *Hadoop YARN*, and *Hadoop MapReduce* [11]. *Hadoop Common* is a basic utility used to support the other modules. *HDFS* provides high-throughput access to the application data. *Hadoop YARN* is the framework for scheduling jobs and manages the clusters. *Hadoop MapReduce*, popular for its large-scale batch processing and high-speed data retrieval, is used for parallel processing of large data sets [11].

Hadoop operates its clusters in a master-slave architecture. The Name Node serves the role as the master and manages the filesystem namespace, and allows access to files requested by the system, including the metadata. There are three major categories of machine roles in Hadoop deployment (Client Machines, Master Nodes, and Slave Nodes) as shown in Figure 4. The Master Nodes are responsible for the HDFS and MapReduce functions. The Name Node coordinates the storage function (HDFS), while the Job Tracker carries out the parallel processing of data using MapReduce. Slave Nodes handle all the machine tasks of storing data and running the computations. Each slave runs both a Data Node and Task Tracker daemon that communicate with the Master Node. The Task Tracker daemon is a slave to the Job Tracker, while the Data Node daemon is a slave to the Name Node. The Name Node retrieves the files, which are divided into one or more blocks and are stored across the Data Nodes. There is typically only one Name Node per cluster which is responsible for reconstructing information and managing the storage. The Name Node knows about the Data Nodes and the Data Node knows about the actual files. The Data Nodes perform all the work of the system, handling blocks when directed by clients or the Name Node. They perform block creation, deletion, and replication directed from the Name Node. Periodically, they report their block information back to the master. The job of the Client machine is to load data into the cluster, create MapReduce jobs describing how the data should be processed, and then retrieve the results once complete.

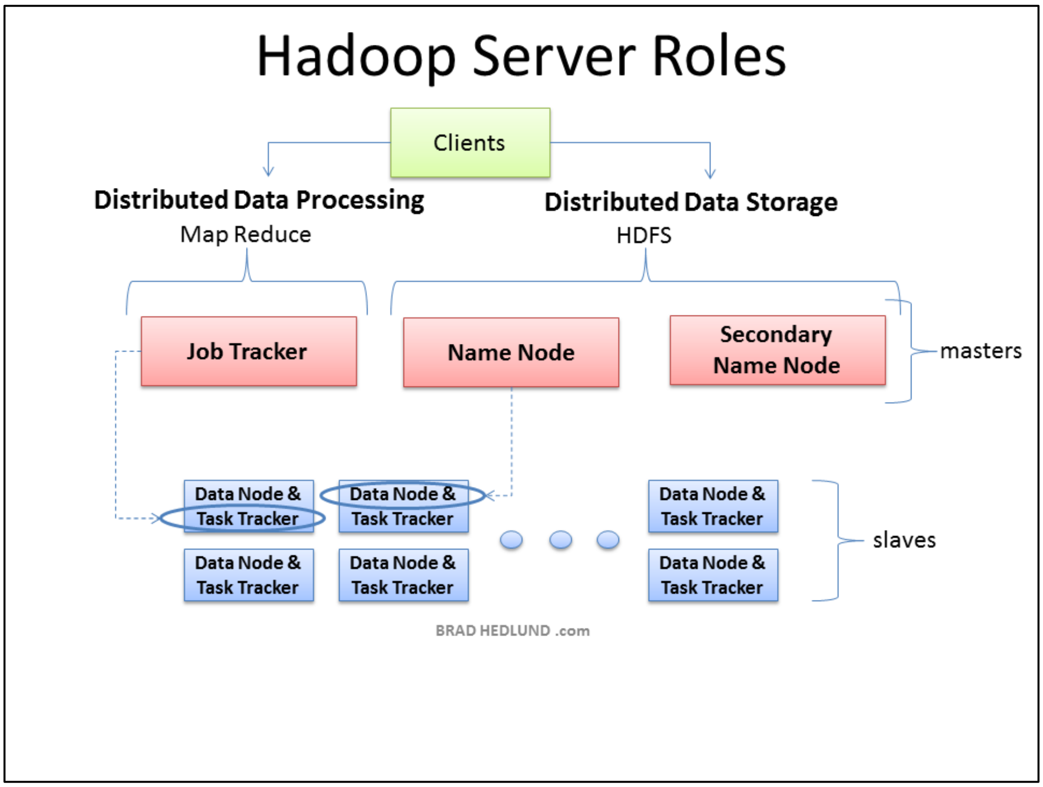


Figure 4. Hadoop Server Roles (from [12]).

As mentioned previously, businesses and governments have a tremendous amount of data that needs to be analyzed and processed very quickly. Hadoop allows them to separate data into smaller chunks, to spread these over multiple machines, and then to process the data in parallel. HDFS is the primary distributed storage used by Hadoop applications and MapReduce is the software framework for writing the applications that process data in parallel across the cluster. These two components are discussed below in further detail.

3. Hadoop Distributed Filesystem (HDFS)

The HDFS manages the storage across a network of machines. It is designed for storing very large files with streaming data access patterns, running on clusters of commodity hardware. The files are hundreds of megabytes, gigabytes, terabytes, or petabytes in size. The design of HDFS is categorized into several attributes: streaming data access, commodity hardware, low-latency data access, lots of small files, and multiple writers (arbitrary file modifications) [13].

4. MapReduce

MapReduce was designed as a distributed data processing model. It divides problems into two parts: a *Map* and a *Reduce* function. MapReduce jobs are split into independent chunks. The Map portion processes the tasks in a parallel manner, while the *Reduce* function sort the outputs of the maps [11]. Map functions can be simultaneously executed without any additional interactions. Storage capacities have clearly increased over the years and the rate at which data can be read from such devices have not been able to keep up. Reading and writing data from a single drive is slower and inefficient. Reducing the reading time by using multiple disks running in parallel offers a powerful paradigm when dealing with large data sets. Hadoop's MapReduce provides a model that overcomes the input/output limitations of disk reading and writing by operating as a batch query processor that sanctions ad hoc queries to be run against datasets in a timely manner [11]. Traditional relational databases with enough disk storage for large-scale batch analysis are not enough to handle big data. MapReduce answers the concern of seek time and is used to address problems that need to analyze the whole dataset in a batch fashion.

5. Hadoop Cluster Process

A Hadoop Cluster needs data and multiple machines working at once to perform fast parallel processing. The client breaks the data into smaller blocks and places the blocks on different machines throughout the cluster. The client communicates each block to the Name Node and receives a list of the Data Nodes that have a copy of the block.

The client then writes the block directly to the other Data Nodes and this is replicated for the other blocks. The Name Node provides the map of where data is and where data should go in the cluster. Hadoop uses a concept called *rack awareness*: the Name Node knows where the Data Nodes are located in the network topology and use that information to make decisions about where data replicas should exist in the cluster [12]. The Nodes communicate using the transmission control protocol (TCP) [12].

The Name Node (see Figure 5) is responsible for the filesystem metadata for the cluster and oversees the health of the Data Nodes. It is the central controller of HDFS, and does not hold data itself. It only knows what blocks make up a file and where those blocks are located in the cluster. Data Nodes send heartbeats to the Name Node at fixed intervals through a TCP handshake, using the port numbers defined for the Name Node daemon. Every tenth heartbeat is a block report, where the Data Node tells the Name Node about all the blocks it has [12]. This number is set by default and can be configured by the administrator. The block report keeps the Name Node current of its metadata and ensures the block replicas exist on different nodes. Without the Name Node, the Clients would not be able to read and write files from HDFS, and it would be impossible to schedule and execute MapReduce Jobs [13]. If the Name Node stops receiving heartbeats from a Data Node, it presumes HDFS is down [12].

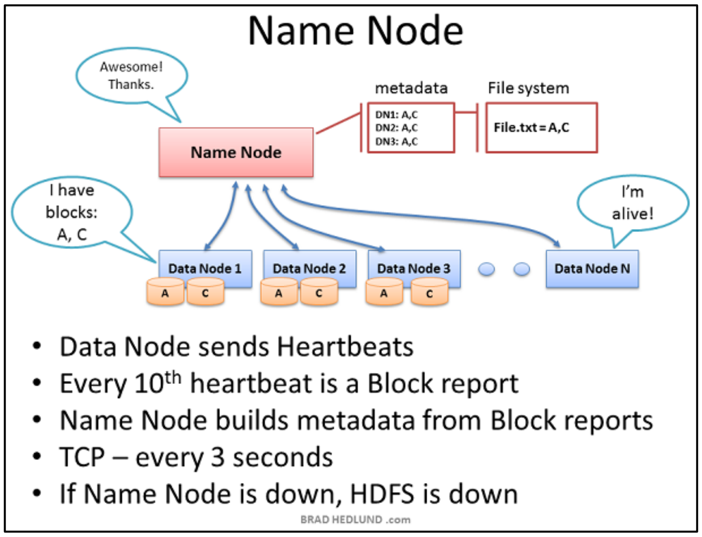


Figure 5. Name Node (from [12]).

Hadoop uses a secondary Name Node that connects to the Name Node to gain a copy of the Name Node's metadata in memory and any files used to store the metadata. The secondary Name Node combines the information in a new file and sends it back to the Name Node, while keeping itself a copy. In the event the primary Name Node fails, the files retained by the secondary Name Node are used to recover the primary Name Node [12].

6. MapReduce: MAP function

The first step of a MapReduce job (see Figure 6), *Map*, is one in which the nodes run some computation on blocks of data local to that node. For example, the node may be instructed to count the number of occurrences of the word "refund" in the data blocks of some file *File.txt*. The client submits this job to the Job Tracker, asking "How many times does 'refund' occur in *File.txt*." The Job Tracker asks the Name Node to learn which Data Nodes hold blocks of *File.txt*. The Task Tracker starts a Map task and monitors the progress [12]. The Task Tracker provides heartbeats and task status back to the Job Tracker. When each Map task completes, each node stores the results of its local computation in the temporary local storage. In the next stage, this data is sent over the network to a node running the Reduce task, to finish the computation.

7. MapReduce: REDUCE function

The second portion of the MapReduce framework is *Reduce*. The map task on the machines have completed and generated their output, now stored in local storage [12]. This data needs to be combined and processed to generate a final result. The Job Tracker starts a Reduce task on any one of the nodes and instructs the Reduce task to retrieve the Map task outputs. Continuing our example, the Reduce task simply sums the occurrences of the word 'refund' and writes the result to a file, *Results.txt* [12]. When complete, the client machine can read the *Results.txt* from HDFS and the job is considered complete.

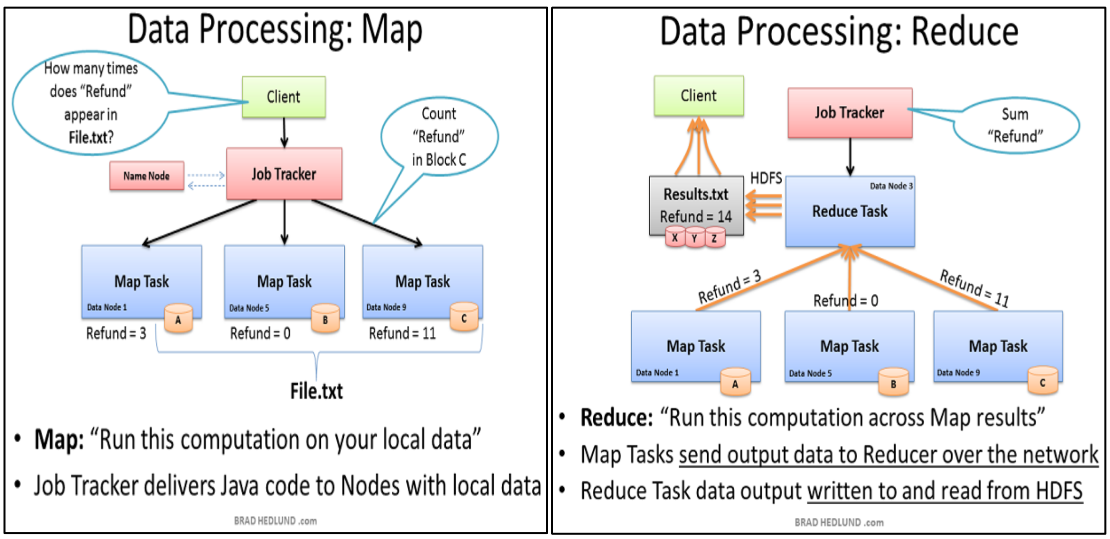


Figure 6. Data Processing: Map and Reduce (from [12]).

C. PROJECT SERENGETI

Serengeti is an open source, virtual appliance (vApp), which acts as a management service to automate the deployment, management and scalability of Apache Hadoop clusters on VMware vCenter platforms. Leveraging the VMware vCenter platform, Serengeti expedites the deployment of a highly available Hadoop cluster, to include common Hadoop components such as HDFS, MapReduce, Pig, and Hive on virtual platforms. In addition, Serengeti has native support for various Hadoop-based distributions, such as Cloudera CDH4, MapR M5, Hortonworks, and Pivotal [4]. Figure 7 represents a high level overview of Serengeti's features.

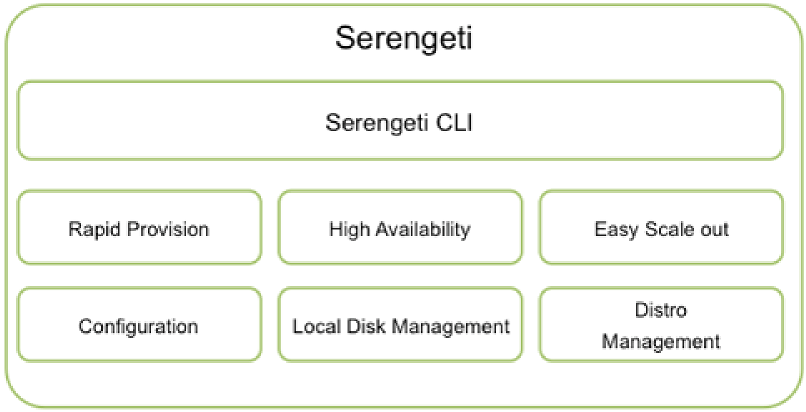


Figure 7. Serengeti Features (from [13]).

1. Serengeti Architecture

The Serengeti vApp runs on top of vCenter and includes a Serengeti Management Server virtual machine and Hadoop Template virtual machine. Figure 8 represents a high level overview of the Serengeti architecture.

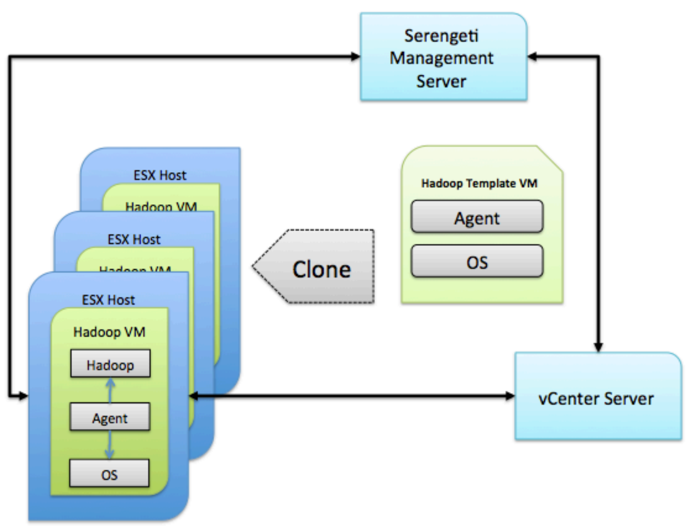


Figure 8. Serengeti Architecture (after [13]).

Serengeti deploys a Hadoop cluster in a number of steps, summarized here from the Serengeti Users Guide. The Serengeti Management Server searches for ESXi hosts with sufficient resources, selects ESXi hosts on which to place Hadoop virtual machines, then sends a request to vCenter to clone and reconfigure virtual machines. The Agent configures the OS parameters and network configurations, downloads Hadoop software packages from the Serengeti Management Server, installs Hadoop software, and then configures Hadoop parameters. Deployment time is significantly reduced because provisioning is performed in parallel [14].

THIS PAGE INTENTIONALLY LEFT BLANK

III. PROJECT OVERVIEW

Our project tasks are divided into four major phases. Table 1 summarizes the status and objectives of each phase. One target configuration to be used in the CISR Big Data Test Bed for experimentation is a Hadoop cluster based on Fedora 13 with SELinux enabled. This configuration would support the development and measurement of experimental Hadoop configurations, such as those described by Nguyen *et al. [15]*. This is the primary motivation for leaving Phase III incomplete, and using Fedora 13 as the target for Phase IV. Each phase is detailed in the chapters that follow.

Phase	Summary	Status	Appendices
PHASE I	Upgrade test bed hardware and software to support Phase II.	Complete	B - H
PHASE II	Use Serengeti to install a Hadoop cluster based on CentOS.	Complete	I
PHASE III	Use Serengeti to install a Hadoop cluster based on Fedora 18.	Incomplete (Tests Fail)	J
PHASE IV	Use Serengeti to install a Hadoop cluster based on Fedora 13.	Complete	J

Table 1. Overview of Project Phases.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. PHASE I

In this phase, the hardware and software of the test bed was upgraded to support the use of vCenter for Serengeti. This included a number of optional hardware upgrades, such as upgrading the vSphere ESXi hypervisor on each host and installing vCenter 5.1. The final test bed setup is summarized in Appendix A, Figure 11. For details of the test bed hardware and server components and their network configuration, see Appendix A.

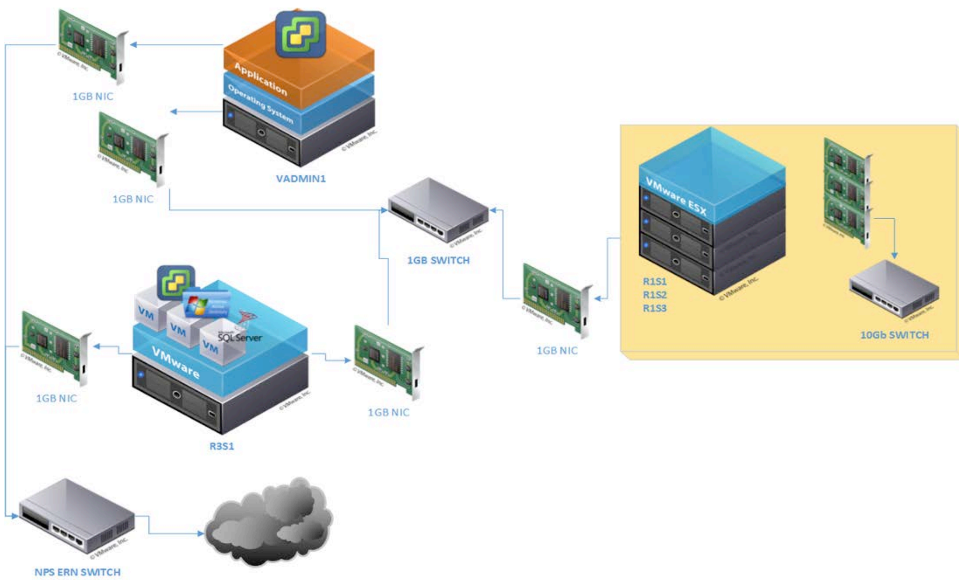


Figure 9. CISR Big Data Test Bed Architecture.

A. HARDWARE UPGRADE

The hardware of the test bed was upgraded with a number of enhancements in order to best leverage performance, reliability, and scalability considerations for the process-intensive production environment. Upgrading the existing test bed server hardware was imperative because it was the single most significant factor that affects the performance of the ESXi hypervisor and the vSphere clients. Our initial focus was to circumvent the anticipated and potential performance bottlenecks associated with CPU, memory, and storage. The most significant upgrades were made on the Dell PowerEdge

R710 servers, which included: six 1TB hard drives, 18 16GB memory DIMMS, and 2-port 10Gb Ethernet network interface cards.

B. UPGRADING TO VSPHERE ESXI 5.1

VMware vSphere ESXi is a bare-metal hypervisor used in the Test Bed (for an overview, refer to Chapter II, Section A.1). We installed vSphere ESXi 5.1 on four hosts; three are part of the production cluster and one is an Administrative server. Prior to installation we referred to the system requirements section of VMware's vSphere 5.1 Installation and Setup guide [16]. We thoroughly reviewed the minimum hardware requirements and the supported server platform compatibility guide. The procedures followed to install VMware vSphere ESXi 5.1 are provided in Appendix B.

C. INSTALLING VSPHERE CLIENT

VMware highly recommends managing ESXi hosts through the vSphere Client or the vSphere Web Client. Both applications offer remote management for the ESXi hosts, vCenter Server, and virtual machines. The vSphere Client eliminates the traditional constraints of centralized management from the physical server console. The ESXi 5.x hypervisor was specifically engineered with remote administration and management as a capability. The vSphere Client is a Windows-specific application interface that provides all of the functionality for managing the virtual infrastructure. The vSphere Web Client is an alternative to the Windows-based vSphere Client; however, it only offers a subset of the functionalities.

The vSphere Client was installed using the vCenter Server installation disk on the VADMIN1 server, then on each subsequent ESXi host. VADMIN1 serves as the main interface for accessing the ESXi hosts. The procedures followed to install the vSphere Client are provided in Appendix G.

D. INSTALLING VCENTER 5.1

The vCenter Server centralizes the management of the ESXi hosts and virtual machines. In preparation for installation of vCenter, various system, network, and database prerequisites had to be met. We created three administrative virtual machines on R3S1 (each using Microsoft Server 2008 R2 as the base operating system) to host each service/component:

- Backend Database Server (Microsoft SQL Server 2008);
- Active Directory, Domain Name System (DNS), and Dynamic Host Configuration Protocol (DHCP);
- vCenter Server.

Figure 10 illustrates the overall VM architecture on R3S1. Installing vCenter Server as a VM affords advantages over using a physical server, including increased portability, snapshot functionality, and cloning functionality [17].

1. Active Directory Support

In order to integrate Active Directory services into vCenter, Active Directory must be installed and properly configured prior to the installation of vCenter. Active Directory was installed on the R3S1 server as a separate Windows VM for user authentication. Active Directory seamlessly integrates in the vCenter architecture with vCenter Single Sign-On. In later steps, during the vCenter Single Sign-On installation, this Active Directory service is selected as the Single Sign-On identity source. The procedures followed to install and configure Active Directory support for vCenter are provided in Appendix D.

2. Backend Database Support

The vCenter Server uses a dedicated backend database server to store logging, statistics, configuration data, permissions, user accounts, and other data. Extensive database configuration is necessary to prepare the vCenter database server. Only after configuring the database can vCenter be installed.; otherwise, the installation will fail. Since the databases' sole purpose is to serve as a repository for vCenter's data, it is paramount to make the database accounts members of the "Domain Admins" group. The vCenter Server explicitly requires full rights to the backend database.

vCenter Server supports IBM DB2, Microsoft SQL Server, and Oracle database servers. We installed Microsoft SQL Server 2008 SP2 on a separate VM hosted on R3S1. The steps performed in configuring the SQL database included: (1) configuring the SQL database to work with vCenter, (2) creating the SQL Server database and the user for vCenter Server, (3) setting database permissions by manual creation of the database roles and the VMware schema, and (4) setting the database permissions. The procedures followed to install and configure backend database support for vCenter are provided in Appendix F.

3. Installing vCenter Server

The vCenter Server suite was installed on its own independent virtual machine hosted on R3S1 in order to leverage specific advantages in the virtual architecture (see Appendix A, Figure 12). For example, as a virtual machine, the vCenter Server can be migrated to another host if needed, or snapshotted for backups and archiving.

Figure 10. Test Bed Administrative VM Infrastructure on R3S1.

The vCenter Server installation proved to be non-trivial, mainly due to the ambiguity of existing documentation and online support at the beginning of this phase. The vCenter Server is the most critical application suite within the entire virtual infrastructure, so it was imperative to ensure that the installation executed as precisely as possible. With vCenter installation can proceed only after the backend database and directory services are installed and configured. The procedures followed to install and configure the vCenter Server are provided in Appendix G.

THIS PAGE INTENTIONALLY LEFT BLANK

V. PHASE II

This phase of the project involves the configuration of the virtual network infrastructure and the installation of Serengeti. Serengeti automates the deployment of Hadoop clusters by cloning a template VM that is installed in vCenter as part of the Serengeti open virtual appliance (OVA) package. The objective of this phase is to deploy a Hadoop cluster in Serengeti using the default supported settings. The default VM template provided for Serengeti 0.8 uses CentOS 5.6 as its operating system. Our reference for completing this objective was the VMware Serengeti User's Guide for Serengeti 0.8, which lists the network and resource requirements to support Serengeti version 0.8, and provides instructions for installing and configuring Serengeti.

A. SETUP VIRTUAL INFRASTRUCTURE

The VMware infrastructure must be configured to support Serengeti operations prior to the deployment of the Serengeti virtual appliance (vApp). The requirements for the installation of Serengeti are listed in Section 2.4 of the VMware Serengeti User's Guide. Serengeti can only be used in a VMware environment where vCenter is installed, which requires a vSphere 5.0 or 5.1 Enterprise license. We met the networking requirements by configuring the Active Directory server built in Phase I to act as the DNS and DHCP server for Serengeti. To facilitate troubleshooting during the next phase of the project, we configured Active Directory to provide Internet connectivity to the Serengeti management server, Hadoop template, and the nodes in the Hadoop clusters created by Serengeti. This step is not required to run Serengeti with its default settings, but it was helpful when performing the software installations during Phase III. The virtual infrastructure configurations we used in this project are documented in Appendix H.

B. INSTALLING THE SERENGETI VIRTUAL APPLIANCE

Serengeti is available as an OVA from the VMware website and is installed using the vSphere client of the vCenter Server. Serengeti can be configured to deploy customized Hadoop clusters, using specific virtual machine settings and software packages. Section 2 of the VMware Serengeti User's Guide [14] lists the available configuration options as well as the different Hadoop distributions supported. Before developing customized configurations, we confirm the basic configuration of the virtual infrastructure, by deploying a Hadoop cluster using the default configurations in Serengeti. This confirms that the infrastructure and vApp are functional; it also serves as a comparison point for customized configurations. The procedures followed for downloading, installing, and confirming functionality of Serengeti are documented in Appendix I.

VI. PHASE III

An important objective of this project is to develop a Serengeti template to support the deployment of Hadoop clusters based on the Fedora 13 operating system. In support of this goal, we needed to develop an understanding of how Serengeti worked and how it is modified to support operating systems other than CentOS 5.6. This phase targets configuring Serengeti to deploy a Hadoop cluster based on Fedora 18, the most current Fedora release at the time of this project. It was believed this would be an appropriate intermediate step, as Fedora 18 is currently better supported than Fedora 13. In this chapter, we report intermediate progress on this task; we note that this phase was left incomplete, as it was not a strong prerequisite for future phases.

There were two major challenges to this phase. The first challenge was the group's inexperience working with Linux-based systems. Fortunately there are many resources available on the Internet that provide information on installing and configuring the various distributions of Linux. The nature of the project required us to perform many tasks in repetition, and as we progressed we became more competent using the Linux command line interface and less reliant on online resources. The second challenge was a lack of documented user experience with Serengeti. There are currently three main sources of information for Serengeti: the GitHub repository that hosts Serengeti's source code, and two Serengeti-related Google Groups forums (serengeti-user and serengeti-dev). Through these, members of the VMware Project Serengeti Team informed us that the current release of Serengeti is not designed to support operating systems other than CentOS and RHEL, but that it may be possible to customize Serengeti to use Fedora by modifying the program's source code; to their knowledge, no one had previously accomplished this.

Our first attempt at modifying the Serengeti vApp was to install the Fedora 18 operating system on the Hadoop template and provision a Hadoop cluster. We made significant strides in this phase of the project, but found that the differences between Fedora 18 and CentOS 5.6 were greater than anticipated. While we moved to Phase IV to deploy a Hadoop cluster using Fedora 13, we did not completely abandon Phase III: we spent approximately four weeks working with both Fedora 13 and Fedora 18 side-by-side. We were first successful in provisioning a Hadoop cluster with the Fedora 13 operating system. Shortly after, we were able to provision a Hadoop cluster with Fedora 18.

The process we used for creating the Hadoop template based on Fedora 18 is the same as the one used for creating a template based on Fedora 13, described in Phase IV of this report. Some of the steps had to be performed differently with the Fedora 18 template because of differences in the operating systems (see Appendix J). Serengeti provides no indication of error while provisioning the Fedora 18 cluster. When Serengeti reports that a cluster has been created successfully, this should be interpreted to mean that the VMs were provisioned and all requested software packages were installed without error. Serengeti does not validate that the Hadoop cluster is functional.

Our secondary functional testing relies on the built-in MapReduce tests provided with Apache Hadoop. Our functional test process is described in further detail in Phase IV of this report. When testing the Hadoop cluster deployed by Serengeti using the experimental Fedora 18 template, communication errors are reported between nodes:

```
java.lang.RuntimeException: java.net.ConnectException:
Call to localhost/127.0.0.1:8020 failed on connection
exception: java.net.ConnectException: Connection refused
```

Additional alterations to the Fedora 18 template may be required in order to create a functional Hadoop cluster, to resolve the above issues. We leave this as future work.

VII. PHASE IV

A. PROCESS OVERVIEW

The objective of this phase is to use Serengeti to provision a Hadoop cluster based on the Fedora 13 operating system. Modifying the default Serengeti template to support Fedora 13 was possible, as it more closely resembled the default operating systems already supported by the template. The key to Phases III and IV of this project was to understand how Serengeti worked and to be able to identify the point of failure when an error occurred during the provisioning process. Serengeti uses vCenter to clone and initialize the VMs. Serengeti then utilizes Chef to install the required software packages used to create the Hadoop cluster. Our research during this phase consisted primarily of creating Hadoop clusters in Serengeti using a Fedora 13 template, then reviewing the standard output log (std.out) to examine failures. We made educated guesses about what configurations to change based on these errors, or acted on feedback from the Project Serengeti Team after posting logs to the serengeti-user forum.

Our first objective was to ensure that we configured our Hadoop template properly so that it could be used by Serengeti to provision the VMs for the Hadoop cluster. The Project Serengeti Team published a guide for creating a Hadoop Template from scratch; however, this guide was designed for the CentOS 5.6 operating system. We found this guide could be applied directly to Fedora 13 with minimal adjustments, because there were significant similarities between CentOS 5.6 and Fedora 13, in terms of filesystem organization and services. The process used to adjust the Hadoop template for use with Fedora 13 is described in Appendix J.

After template configuration, we modified the Chef cookbooks to control how Serengeti configured each Fedora machine deployed in the Hadoop cluster. We found only two cookbook modifications were required for Serengeti to successfully complete installation of all required software packages on the Fedora 13 operating system. The process for modifying the cookbooks is outlined in Appendix K.

B. BENCHMARK TESTING

Once we were able to successfully provision Hadoop clusters in Serengeti, we conducted tests to determine functionality and performance. We used the VMware white paper, “A Benchmarking Case Study of Virtualized Hadoop Performance on VMware vSphere 5”[1], as an example of how to benchmark Hadoop clusters. This document describes three types of Hadoop tests that can be used to measure the performance of a Hadoop cluster: Pi, TestDFSIO, and Terasort. Of the three tests, Terasort is considered to be the most accurate representation of an actual Hadoop workload [18]. We used Terasort, TeraGen and TeraValidate to test our clusters.

For further explanation and the correct syntax for running these tests, we referenced a blog entry by Michael Noll, titled “Benchmarking and Stress Testing a Hadoop Cluster With Terasort, TestDFSIO & Co.” [19]. We provisioned Fedora 13 Hadoop clusters with 3, 10, and 25 worker nodes and CentOS 5.6 Hadoop clusters of 3 and 10 Hadoop clusters. The following tables show the results of these tests. The times are listed in seconds.

CentOS 5.6 Cluster (1 master, 1 client, 3 workers)				
	1GB	10GB	100GB	1TB
TeraGen	40.61	1593.89	FAILED	FAILED
TeraSort	276.73	3284.64	N/A	N/A
TeraValidate	40.71	178.32	N/A	N/A

Table 2. CentOS 5.6 Cluster with 3 Worker Nodes (time in seconds)..

Fedora 13 Cluster (1 master, 1 client, 3 workers)				
	1GB	10GB	100GB	1TB
TeraGen	37.87	314.74	FAILED	FAILED
TeraSort	79.19	1919.73	N/A	N/A
TeraValidate	35.83	198.36	N/A	N/A

Table 3. Fedora 13 Cluster with 3 Worker Nodes (time in seconds)..

[1] Available at http://www.vmware.com/files/pdf/VMW-Hadoop-Performance-vSphere5.pdf.

CentOS 5.6 Cluster (1 master, 1 client, 10 workers)				
	1GB	10GB	100GB	1TB
TeraGen	67.33	330.43	FAILED	FAILED
TeraSort	481.02	4200.49	N/A	N/A
TeraValidate	68.63	20.46	N/A	N/A

Table 4. CentOS 5.6 Cluster with 10 Worker Nodes (time in seconds)..

Fedora 13 Cluster (1 master, 1 client, 10workers)				
	1GB	10GB	100GB	1TB
TeraGen	21.05	32.56	1770.76	FAILED
TeraSort	41.18	101.42	3031.05	N/A
TeraValidate	29.24	41.40	183.70	N/A

Table 5. Fedora 13 Cluster with 10 Worker Nodes (time in seconds)..

Fedora 13 Cluster (1 master, 1 client, 25 workers)				
	1GB	10GB	100GB	1TB
TeraGen	1876.06	23576.26	FAILED	FAILED
TeraSort	2932.62	FAILED	N/A	N/A
TeraValidate	180.85	N/A	N/A	N/A

Table 6. Fedora 13 Cluster with 25 Worker Nodes (time in seconds)..

As seen by the results of these tests, we had limited success with the performance of our Hadoop clusters. Additionally, while we noticed an improvement in performance when increasing the number of workers from 3 to 10, our performance suffered significantly when we increased our cluster size to 25 workers. We were not able to determine the reason for this degradation. When running tests with 25 workers we observed CPU utilization alarms on the ESXi hosts despite not having limits set on the CPU reservations. The amount of time spent on determining proper configurations for the template and the Chef cookbooks prevented us from dedicating sufficient time to troubleshooting these performance issues. It is recommended that these issues are investigated in future research.

THIS PAGE INTENTIONALLY LEFT BLANK

VIII. SUMMARY AND CONCLUSION

A. SUMMARY

In this Capstone project we determined that Serengeti can be configured to provision Hadoop clusters using the Fedora operating system. There are many requirements that must be met in order to accomplish this, such as obtaining appropriate software licenses, meeting hardware requirements, building the vCenter infrastructure, and configuring network services to facilitate communication between Serengeti and vCenter. The ability to automate Hadoop clusters may serve a valuable purpose in future research at the Naval Postgraduate School; however, more research is needed to determine whether Serengeti can be used to provision MLS-aware Hadoop clusters.

B. RECOMMENDATIONS FOR FUTURE WORK

There are further areas of research necessary in order to determine the usefulness of Serengeti and the Big Data Analytics test bed. One area of study should address the performance issue with the Hadoop clusters. In most attempts, we were unable to complete a Terasort test greater than 10 GB. When scaling out the clusters to a size greater than 10 worker nodes, running tests will result in CPU usage alarms on the ESXi hosts. We spent time researching this problem, but did not find a solution.

If we were to continue this project, the next phase would be to provision a Hadoop cluster with SELinux enabled. The current Serengeti configuration requires SELinux to be disabled; however SELinux is required for MLS-aware Hadoop. Further research is required to determine if Serengeti can be configured to provision VMs with SELinux enabled. The final phase of the project would be to provision an MLS-aware Hadoop cluster in Serengeti.

If funding becomes available to upgrade the Dell PowerEdge 2950 (R3S1), it is recommended that this server be replaced with a model that supports a 10Gb NIC. This server hosts the vCenter administrative VMs (vCenter, AD, and SQL). A 10Gb switch was purchased for this project to increase data transfer rates between the ESXi hosts;

however, we were unable to take advantage of this because the Dell PowerEdge 2950 only supports a 1Gb NIC. A 10Gb connection between Serengeti and vCenter should improve cluster provisioning speeds and possibly improve network communication between Hadoop nodes.

APPENDIX A. OVERVIEW OF TEST BED CONFIGURATION

The CISR Big Data Test Bed is comprised of two Dell PowerEdge 42U racks (see Figure 11). Rack 1 contains: (3) Dell PowerEdge R710 servers (R1S1-R1S3), (1) 24-port Gb Cisco 3560 switch, (1) 144-port Gb Cisco 6504 switch, (1) Keyboard/Video/Monitor (KVM), and (1) APC Smart-UPS 3000. Rack 2 contains: (2) Dell PowerEdge 2950 servers, (1) KVM, (1) APC Smart-UPS 3000, and (1) APC Smart-UPS 2200. The hardware configurations of these devices are summarized in Table 7.

CISR Big Data Testbed Rack Configuration

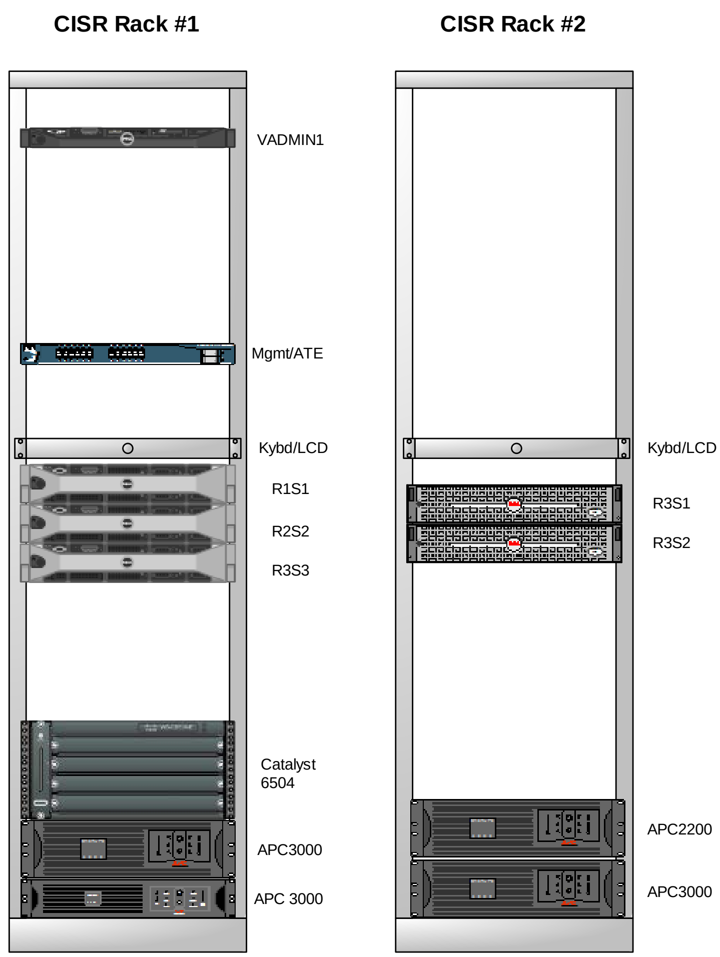


Figure 11. Rack Diagram for the Test Bed.

Name	*Server*	*RAM*	*HDD*	*CPU*	*NIC*	*OS*
R1S1	Dell PowerEdge R710	288 Gb	8TB	X5570 Xeon Processor, 2.93 GHz 8M Cache	(2) 10 GB	VMWare ESXi 5.1.0 (VMKernel Release build 799733)
R1S2	Dell PowerEdge R710	288 Gb	8TB	X5570 Xeon Processor, 2.93 GHz 8M Cache	(2) 10 GB	VMWare ESXi 5.1.0 (VMKernel Release build 799733)
R1S3	Dell PowerEdge R710	288 Gb	8TB	X5570 Xeon Processor, 2.93 GHz 8M Cache	(2) 10 GB	VMWare ESXi 5.1.0 (VMKernel Release build 799733)
R3S1	Dell PowerEdge 2950	63 Gb	8TB	X5570 Xeon Processor, 2.93 GHz 8M Cache		VMWare ESXi 5.1.0 (VMKernel Release build 799733)
VADMIN	Dell PowerEdge R300	136 Gb	4GB	Dual 3.00 Ghz Intel XEON (R)		Windows 2008 Std Svr
Dell PowerConnect 8024F	Switch					
Cisco Catalyst 3560G	Switch					Cisco IOS 12.2(35) SE5 (SW Image: C3560-IPBASE-M)

Table 7. Overview of Server Configurations in Test Bed.

C. POWER

Each rack is equipped with two uninterruptable power supply units. All server systems and network gear are powered from the uninterruptible power supplies (UPS), with the exception of the Cisco Catalyst 6500 for which one power supply is attached to an UPS and the other is directly attached to the lab's conditioned power.

D. NETWORK

Figure 12 illustrates the test bed's physical and virtual network topology. The network is segmented into two subnets, the Management Network and the Campus LAN. The Management Network is on the 10.10.0.0/16 subnet and serves as the primary subnet for the test bed. The Campus LAN is on the 172.20.0.0/16 subnet, which connects to the campus WAN. The Campus LAN is used for the purpose of Internet connectivity in order to install software updates and patches, hence, the reason why we dual-homed most of the servers (VADMIN1, R3S1, and the VMs).

The management network uses the 1Gb Dell PowerConnect 8024F backbone switch and the Campus LAN uses the 1Gb Catalyst 3670 backbone switch. The Management Server (VADMIN1) and our production servers (R1S1-R1S3, R3S1-R3S2) are physically connected to each of these switches. The physical network allows for connectivity to take place between the physical machines which host the virtual machines. VMware ESXi 5.1 is installed on all the servers server's with the exception of the VMs. Three virtual machines (vCenter Server, Active Directory/DNS Server, and MS SQL 2008 Server) are installed on R3S1, and R1S1-R1S3 are used as the production servers to host the Hadoop Clusters.

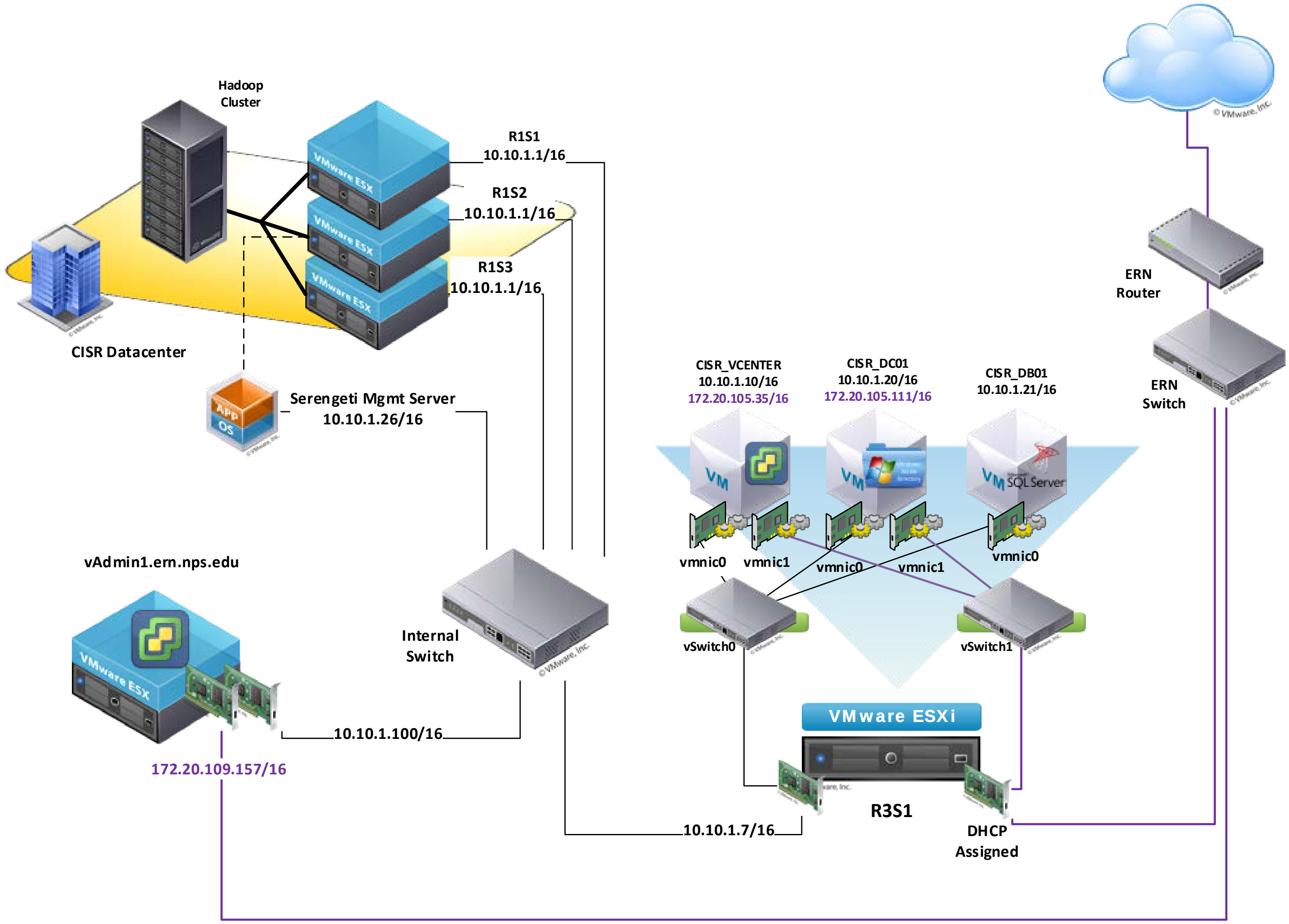


Figure 12. Overview of the Test Bed Network Topology.

E. CONTROL

Each rack has keyboard, video and mouse (KVM) to control the systems in that rack. Both racks each have an administration system from which all the ESXi host systems are controlled. The vSphere Client and vSphere Web Client are installed on VADMIN1 for the purpose of accessing all of the ESXi hosts directly or indirectly through vCenter. The Web Client offers an alternative means to connect to the ESXi hosts through a web browser.

1. Access

There are a number of specific user accounts for different services, applications, and traditional local/domain level access. From vSphere, in order to login directly to an ESXi host, the administrator must use the `root` account to gain access then traverse to the desired VM. The local/domain user accounts are annotated in Table 8, along with all other necessary accounts.

Access	Username	Password
vSphere	Root	(default password 1)
R1S2	Root	(default password 1)
R1S3	Administrator	(default password 1)
R3S1	Administrator	(default password 1)
R3S2	Administrator	(default password 1)
Domain Admin	Administrator	(default password 1)
Local Admin	Administrator	(default password 1)
Single Sign-On DB	RSA_User DBA_User	(default password 1) (default password 1)
vCenter	vCenter_user	(default password 1)
SQL SERVER User	sql_user	(default password 1)
Serengeti Mgmt Svr	Serengeti	(default password 2)
Hadoop Cluster	cpo365	(default password 2)

Table 8. Server/Application Login Credentials.

F. SUBNETWORKS

There are two subnets in the test bed: the Management Network and the Campus LAN. The identities of each server are summarized in Table 9.

KVM	Server	Management Network	Campus LAN	Important services
1	VADMIN1	10.10.1.100/16	DHCP Assigned	
1	R1S1	10.10.1.1/16	Not In Use	
1	R1S2	10.10.1.2/16	Not In Use	
1	R1S3	10.10.1.3/16	Not In Use	
2	R3S1	10.10.1.7/16	DHCP Assigned	
2	R2S2	Not In Use	Not In Use	
	CISR_VCENTER	10.10.1.10/16	DHCP Assigned	SSO, vCenter
	CISR_DC01	10.10.1.20/16	DHCP Assigned	DNS, DHCP, AD DS
	CISR_DB01	10.10.1.21/16	Not In Use	MS SQL

Table 9. Network Identities of Servers in the Test Bed.

APPENDIX B. INSTALLING ESXI HOST

VMware offers several options for installing ESXi, depending on the range of deployment sizes in each specific environment. There are three approaches available for ESXi deployment: (1) ESXi Installable, (2) ESXi Embedded and (3) ESXi Stateless. We opted to use the standard option (ESXi Installable); this allows the hypervisor to be installed on bare-metal hardware, including USB flash drives and/or SD cards mounted on the server. The steps are outlined below.

A. PREPARE BOOT CD

The ESXi Installer ISO image was downloaded from the VMware website[2] and a bootable CD was created (see Figure 13).

```
VMware-VMvisor-Installer-5.1.0–799733.x86_64.iso
File size: 301M
File type: ISO
Release Date: 2012–09–10
Build Number: 799733
ESX i5.1 ISO image (Includes VMware Tools)
```

Figure 13. ESXi Installation ISO used in Test Bed.

B. INSTALL PROCEDURE

The interactive installation mode is a simplistic text-based installer that is fairly easy to walk through (see Figure 14). Through the interactive install, the installer boots into the ESXi text installer and prompts the administrator for installation to a local host disk. The installer reformats and partitions the target disk and installs the ESXi boot image. All previous data located on the drive are overwritten, including hardware vendor partitions, operating system partitions and associated data.

[2] Available online at: https://my.vmware.com/web/vmware/details?productId=285&downloadGroup=VCL-VSP510-ESXI-510-EN.

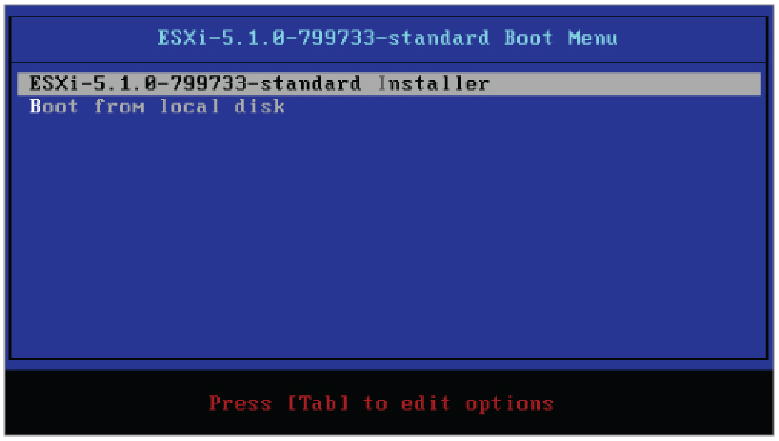


Figure 14. ESXi Text Installer (from [20]).

vSphere ESXi 5.1 comes with a 60-day evaluation-mode trial license. We obtained a license through the VMware website by setting up and activating an account. From there, we were able to register our license. The licensing information is managed through vSphere Client. Strictly adhering to the installation and setup guide, the ESXi hypervisor was installed on the embedded internal USB drive for R1S1, R1S2, and R1S3. On the R3S1 server, the ESXi hypervisor was on the internal Hard Drive. Once the installation was complete, static IP addresses were assigned to each server.

C. ESXI CONFIGURATION

When rebooting the ESXi host for the first time or after resetting the configuration defaults, the host enters an auto-configuration phase. This phase configures initial system and network configuration parameters such as: setting the administrative password, system log management, system services, remote access, etc. By default, DHCP configures the host IP address, and all visible blank internal disks are formatted with the Virtual Machine File System (VMFS) so that virtual machines can be stored on the disks [16].

The Direct Console User Interface (DCUI) is used for ESXi configuration and troubleshooting. We configured the management network and root password through the DCUI (see Figure 15). The DCUI automatically initiates and displays after the auto-configuration phase completes, if a keyboard and monitor is connected to the host. The default behavior is to configure the ESXi management network using DHCP. You will need to override the default behavior and use static IP settings for the management network after the installation is complete. We accessed the "Configure Network Management" option in the DCUI and manually added the assigned IP address to each host, to include the default gateway (10.10.1.1) and subnet mask (255.255.0.0). We also configured the Primary DNS Server (10.10.1.20) and the Hostname (r1s1), and the DNS Suffix (mysea.cisr).

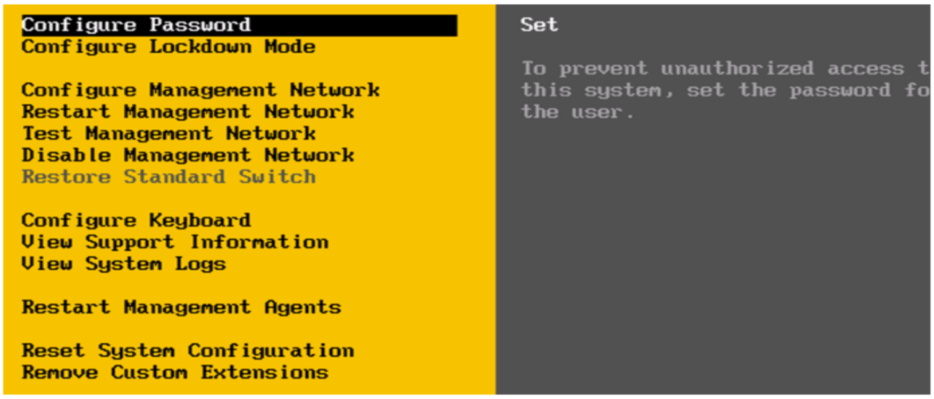


Figure 15. Direct Console User Interface (from [20]).

ESXi hosts can be managed remotely by installing vSphere Client, vSphere Web Client, and vCenter Server. We configured our hosts so that they could be remotely managed by vSphere Client. The vSphere Client serves has a management server with network access to the ESXi hosts. Traffic between the ESXi hosts and vSphere Client or vCenter Server is transmitted via the Ethernet network adapter on the host.

D. TESTING

Following installation, the management network was tested through the DCUI. In DCUI, we accessed "Test Management Network." By default, the performed test attempts to ping the default gateway which is represented as *ping address #0*, DNS Servers, *ping address#1*, and resolves the hostname, *r1s1.mysea.cisr*. After the installation and configuration of the ESXi hosts, vSphere Client was installed for all remote host management.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. CREATING VIRTUAL MACHINE

Virtual Machines can be created via the vSphere Client directly on the ESXi server or the vSphere Web Client interface. Throughout our project, we used the vSphere Client as per the vSphere 5.1 Documentation Guide.[3]

A. CREATE NEW VM

This procedure is used for creating the administrative VMs for Active Directory and the SQL Database (Appendices D and F).

1. From the vSphere Client, connect to the vCenter Server as an `Administrator` (see Appendix A, Table 8).
2. Click on the VMs and Templates icon from the inventory Home.
3. Expand the selection, right-click on the CISR Data Center then select **New Virtual Machine**. Select **Typical** then click **Next**.
4. Enter *AD/DNS/DHCP* or *SQL VM* as the name of the new VM and select *datastore1*as the destination storage for the VM files. Click **Next**.
5. Select *Windows* as the Guest Operating System, then select *Microsoft Windows Server 2008 R2 (64-bit)* as the version. Click **Next**.
6. Designate *2* NICs to connect. For NIC 1, choose *VM Network* and for NIC 2, choose *CAMPUS LAN*. Leave adapters set to *E1000* and ensure the "Connect at Power On" boxes remain checked. Click **Next**.
7. Designate *200GB* as the Virtual disk size and choose *Thin Provision*. Click **Next**.
8. Review new VM settings. If satisfied, click **Finish**.

B. WINDOWS SERVER 2008 R2 STANDARD INSTALLATION

1. Obtain the Windows 2008 Server R2 ISO file.
2. Upload ISO file to the datastore and Map to VM (see Appendix L, sec. B-C).
3. Log into the VM.

[3] Available online at: http://pubs.vmware.com/vsphere-51/index.jsp#com.vmware.vsphere.vm_admin.doc/GUID-0433C0DC-63F7-4966-9B53-0BECDDEB6420.html.

4. Refer to Techotopia's installation instructions[4] for the step-by-step GUI installation wizard.

[4] Performing a clean install of Windows Server 2008 R2. Available online at: http://www.techotopia.com/index.php/Performing_a_Clean_Windows_Server_2008_R2_Installation#Starting_the_Installation_Process.

APPENDIX D. INSTALLING AND CONFIGURING ACTIVE DIRECTORY SUPPORT FOR VCENTER

Any server hosting Active Directory Domain Services (AD DS) is an Active Directory Domain controller. AD DS is a repository for directory data, manages all communication between the users and domains, to include: user logins, authentication, and directory searches [21]. AD DS was installed on the test bed on a VM on R3S1, per the below procedure.

A. INSTALL AND CONFIGURE ACTIVE DIRECTORY DOMAIN CONTROLLER

Do the following:

1. Create a new VM (see Appendix C).
2. Log on to the AD/DNS/DHCP VM.
3. Refer to the Microsoft TechNet instructions[5] for guidance on installing AD DS. Use selections that match Figure 16 and Figure 17.

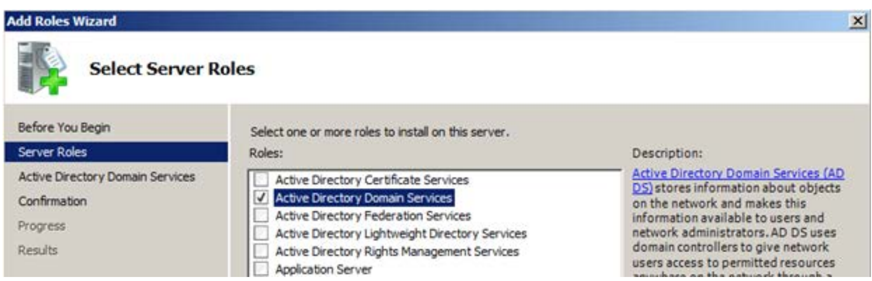


Figure 16. Add Roles Wizard – Server Roles Selection.

[5] "Installing a New Forest by Using the Graphical User Interface," available online at: http://technet.microsoft.com/en-us/library/cc755059(v=ws.10).aspx.

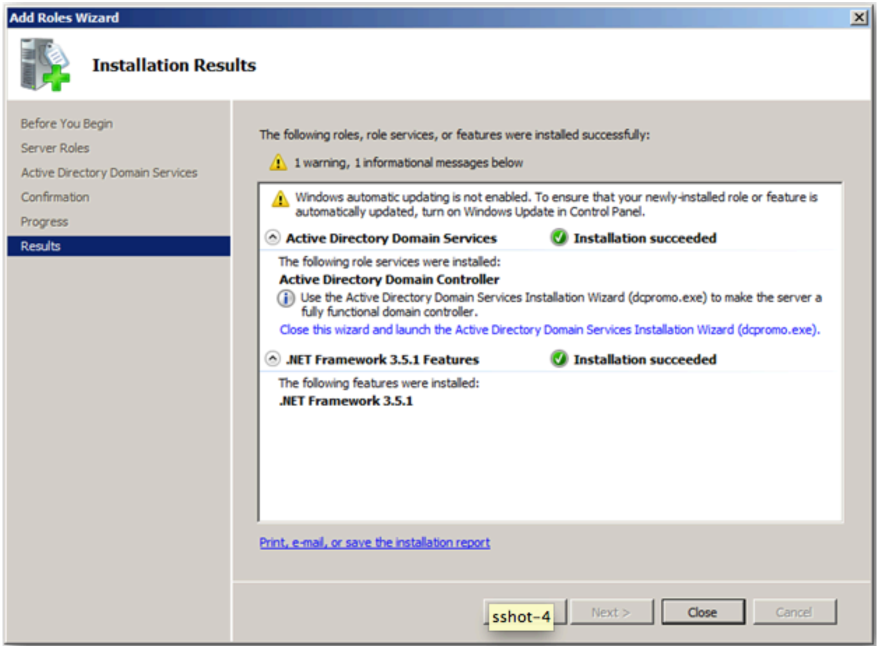


Figure 17. Add Roles Wizard – Installation Results Summary.

APPENDIX E. ADDITIONAL CONFIGURATIONS

In order to ensure sufficient functionality we had to deploy DHCP server and Network Time Protocol (NTP) server. These services were necessary in order to enable the required dynamic allocation of IP addresses, and to ensure proper time sync across all ESXi hosts. Time sync is essential for proper VMware ESXi operations. We enable Internet access to our internal network, by configuring our internal DNS server to forward DNS queries to the CAMPUS DNS server, we set **CISR_DC01** as dual-homed machine and enabled Network Address Translation (NAT) and routing services on it. Enabling Internet access to the internal network allowed us to conduct research and run updates more effectively on all internal nodes including the newly provisioned Hadoop nodes.

A. CONFIGURE AD DC AND VCENTER SERVER AS DUAL-HOMED SERVERS.

Do the following:

1. Log on **VADMIN1** server as an `Administrator` (see Appendix A, Table 8).
2. Double click on **VMware vSphere Client** icon on the Desktop and login to **R3S1** using `administrative` credentials (see Appendix A, Table 8).
3. Expand the host on the left pane, then right-click on the **AD/DNS/DHCP VM**. Select **Edit Settings (**see Figure 18).

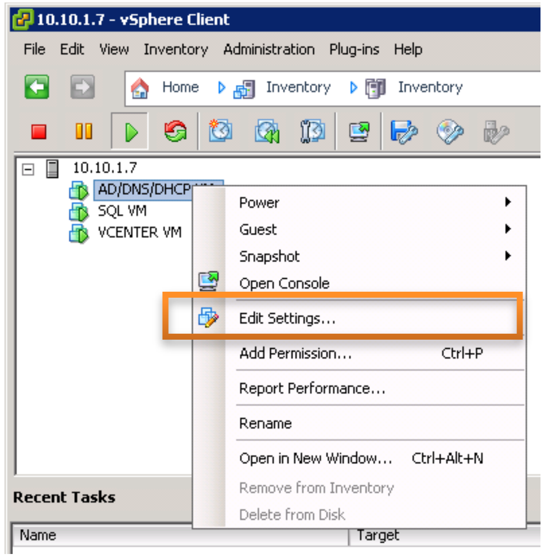


Figure 18. Virtual Machine Edit Settings Menu Option.

4. Click on the **Hardware** then Click **Add**.
5. Highlight **Ethernet Adapter** then click **Next.**
6. Configure **Adapter Type**: *E1000*, **Network Label**: *CAMPUS NET* (see Figure 19).
7. Ensure that **Connect at power** is checked (see Figure 19).
8. Click **Next** > **Finish.**

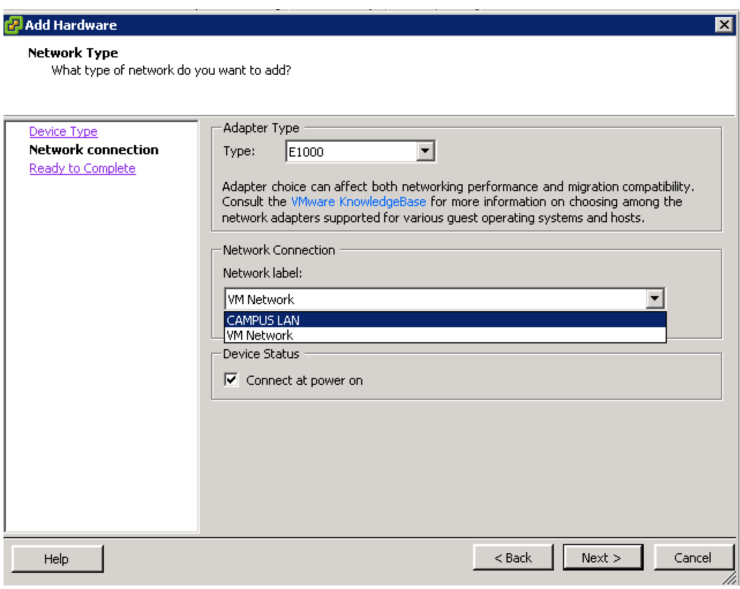


Figure 19. Add Hardware Page.

9. Repeat steps 3 thru 8 for the **VCENTER VM.**

B. CONFIGURE NETWORK ADAPTERS

1. Log on **VADMIN1** server as an `Administrator` (see Appendix A, Table 8).
2. Double click on **VMware vSphere Client** icon on the Desktop and login to **R3S1** using `Administrator` credentials (see Appendix A, Table 8).
3. Expand the host on the left pane, then right-click on the **AD/DNS/DHCP VM.** Select **Open Console.**
4. On the Menu bar click on **VM** > **Guest** > **Send Ctrl+Alt+del** (see Figure 20).

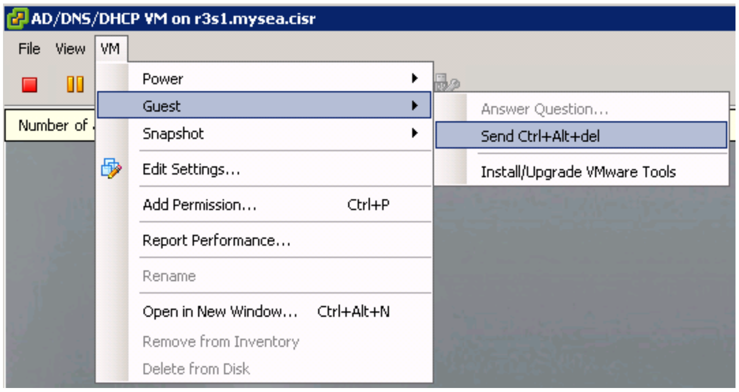


Figure 20. Send Ctrl+Alt+Del Menu Option.

5. On the Menu bar click on **Inventory** > **Virtual Machine** > **Guest** > **Send+Ctrl+del.**
6. Log on the server using `Administrator` credentials (see Appendix A, Table 8).
7. Click on **Start** > **Control Panel** > **Network and Sharing Center** > **Change Adapter Settings.**
8. Right click on **Local Area Connection** and select **Properties** (see Figure 21).
9. Uncheck **Internet Protocol Version 6 (TCP/IPv6).**
10. Highlight **Internet Protocol Version 4 (TCP/IPv4)** and click on **Properties.**
11. Verify all settings in accordance with Appendix A.
12. Click **OK** > **Close.**
13. Right click on **Local Area Connection.**
14. Select **Rename** and rename to: **INTERNAL LAN.**

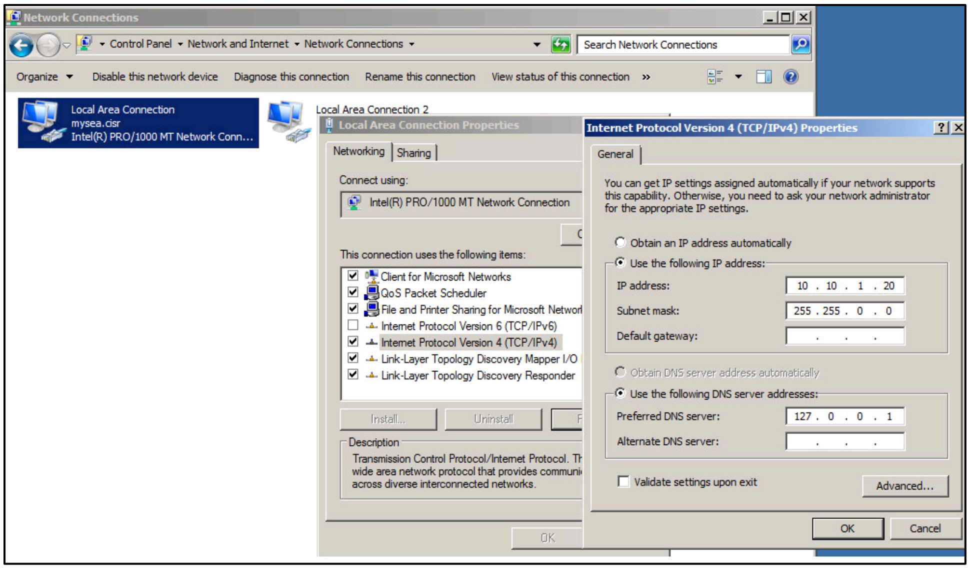


Figure 21. Local Area Connection 1 IPv4 Properties Page.

15. Right click on **Local Area Connection 2** and click on **Properties** (see Figure 22).

16. Uncheck **Internet Protocol Version 6 (TCP/IPv6).**

17. Highlight **Internet Protocol Version 4 (TCP/IPv4)** and click on **Properties.**

18. Verify **Obtain an IP address automatically** and **Obtain DNS server addresses** are checked.

19. Click **OK > Close.**

20. Right click on **Local Area Connection.**

21. Select **Rename** and rename to: **CAMPUS LAN.**

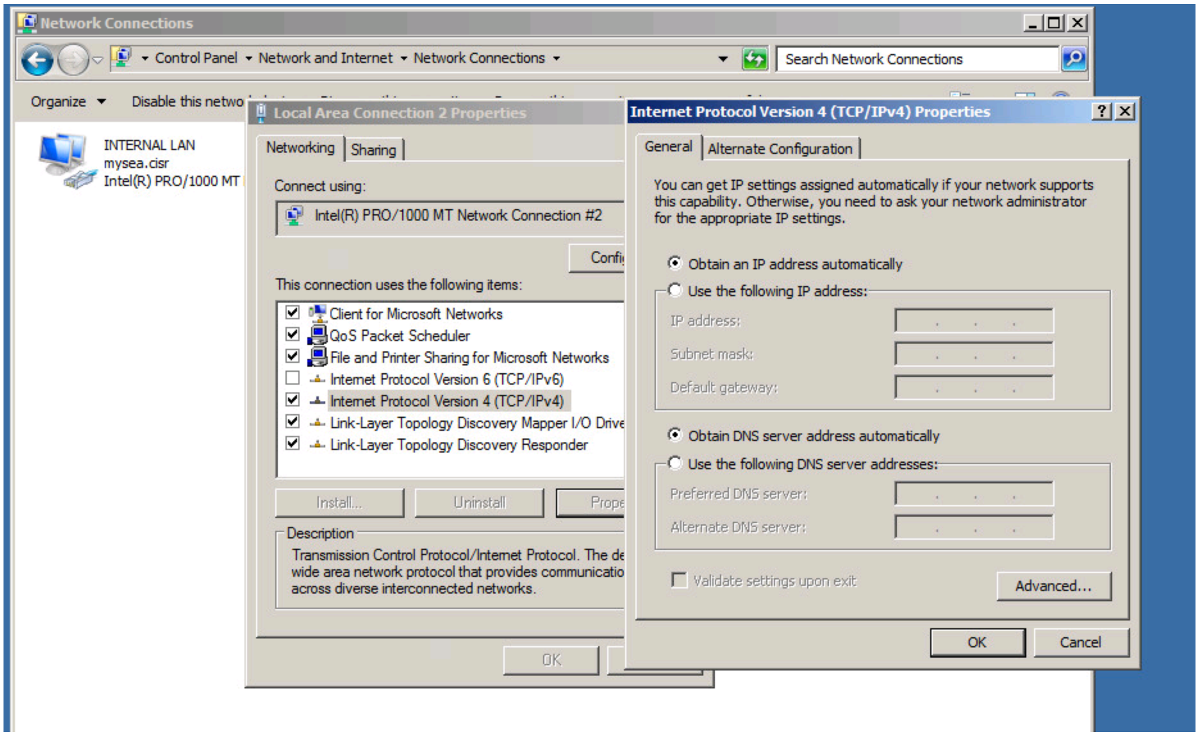


Figure 22. Local Area Connection 2 IPv4 Properties Page.

C. CONFIGURE INTERNAL DNS

1. Click on **Start** > **Administrative Tools** > **DNS.**
2. Highlight **CISR_DC01** and click **Properties.**
3. Click on the **Forwarders** tab, click edit and enter the IP addresses of external DNS servers (see Appendix A, Table 9).
4. Click **OK** twice.
5. Close DNS Manager.

D. DEPLOY AND CONFIGURE A DHCP SERVER

1. Click **Start** > **Administrative Tools** > **Server Manager.**
2. In the left pane, right click on **Roles** > **Add Role** and check **DHCP** (see Figure 23**)**.

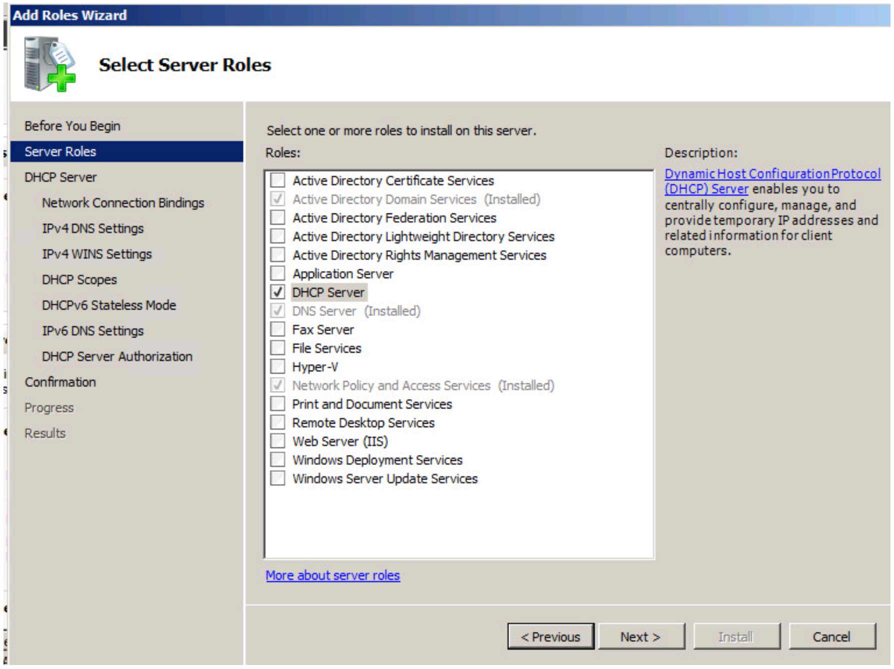


Figure 23. Add Roles Wizard.

3. Click **Next > Next > Next.**
4. Enter the IP address of the internal DNS server (Appendix A, Table 9) under **Preferred DNS server IPv4 address** and click **Next > Next.**
5. Click on **Add** then enter (see Figure 24):

- **Scope name**: `SERENGETI`
- **Starting IP address** `10.10.1.30`
- **Ending IP address** `10.10.1.253`
- **Subnet type:** Wired (lease duration will be 8 days)
- Check **Activate this scope**
- **Subnet mask:** `255.255.0.0`
- **Default gateway:** `10.10.1.20`

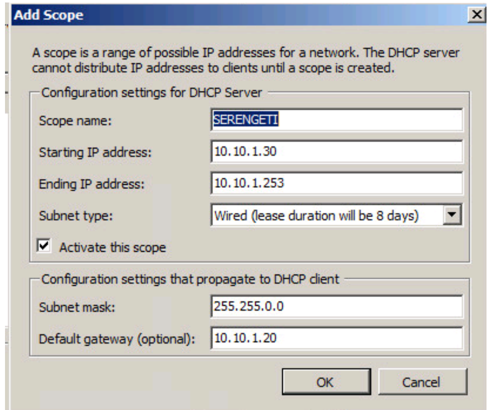


Figure 24. DHCP Add Scope Page.

5.1. Click **OK** > **Next.**
5.2. Check Disable **DHCPv6 stateless mode for this server.**
5.3. Click on **Next** > **Next** > **Install.**
5.4. After server reboot, logon again using admin credentials.
5.5. Click on **Start** > **Administrative tools** > **DHCP.**
5.6. Expand **cisr_dc01.mysea.cisr.**
5.7. Expand **IPv4.**
5.8. In the right pane double-click on **Address Pool,** right-click in the right pane, select **New Exclusion Range** and enter:

- **Start IP address:** `10.10.1.100` (`VADMIN1`, Appendix A, Table 9)
- **End IP address:** `10.10.1.100` and click **OK.**

E. DEPLOY NTP SERVER

1. Configure NTP server on accordance with Microsoft's Authoritative Time Server setup instructions.[6]

F. CONFIGURE ROUTING AND INTERNET CONNECTIVITY

1. Click **Start** > **Administrative Tools** > **Server Manager.**
2. In the left pane, right click on **Roles** > **Add Role,** check **Network Policy and Access Services** and click **Next** > **Next.**
3. Check on **Routing and Remote Access Services** (see Figure 25)**.**

[6] Microsoft Knowledge Base, How to configure an authoritative time server in Windows Server. http://support.microsoft.com/kb/816042.

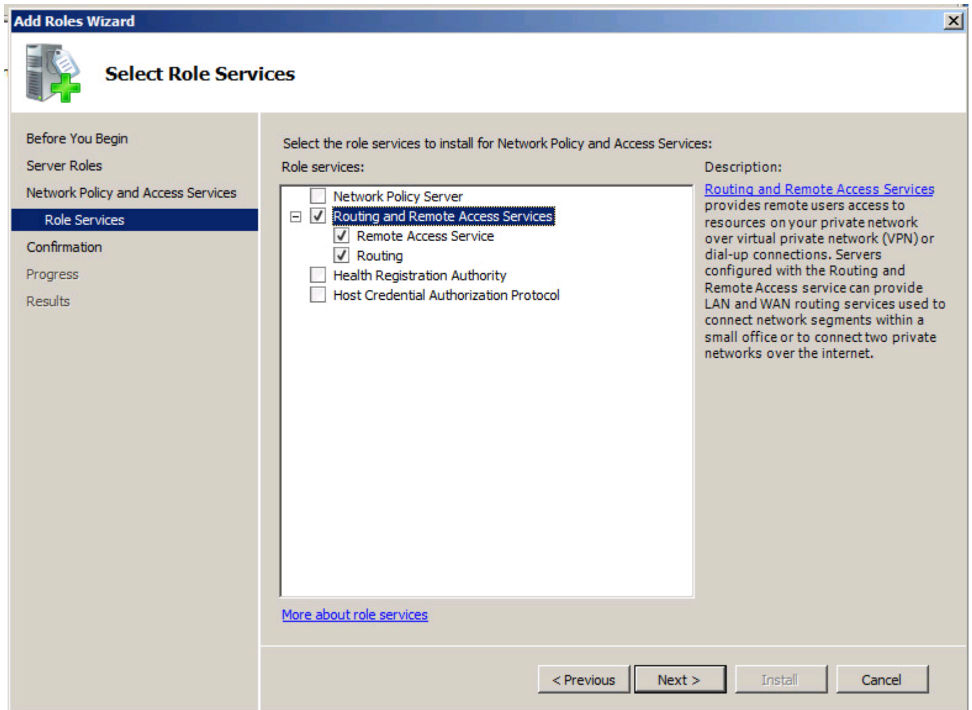


Figure 25. Add Roles Wizard Page.

4. Click **Next > Install > Close.**
5. At the Server manager snap-in, in the left pane expand **Network Policy and Access.**
6. Expand **Routing and Remote Access.**
7. Expand **IPv4** (see Figure 26)**.**

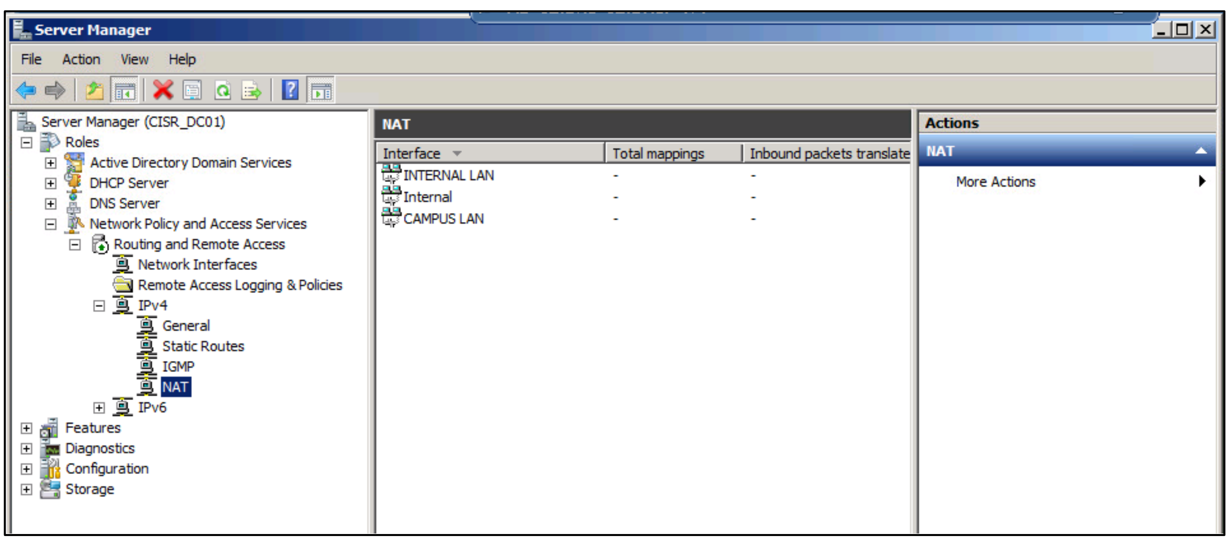


Figure 26. Server Manager NAT Settings.

7.1. Right click on **NAT > New Interface.**
7.2. Select INTERNAL LAN and ensure **Private interface connected to private network** is checked.
7.3. Click **OK.**
7.4. Right click on **NAT > New Interface.**
7.5. Select CAMPUS LAN and check the **Public interface connected to Internet** option.
7.6. Ensure **NAT** is checked then click **OK.**

Note: Ensure the Antivirus/Firewall application does not block NAT traffic

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX F. INSTALLING AND CONFIGURING BACKEND DATABASE SUPPORT FOR VCENTER

These installations assume that Active Directory services have been installed and configured (see Appendix D).

A. INSTALL MICROSOFT SQL SERVER 2008 R2 STANDARD

Do the following:

1. Set up `sql_user` account.
   1.1. Log on to *CISR_DC01*, and launch **Active Directory Users and Computers**.
   1.2. From the main menu, click **Action**, then select **New User**.
   1.3. Create a new Active Directory user account `sql_user`. Choose a password for the user, and add the user to the "Domain Users" group (see Figure 27).

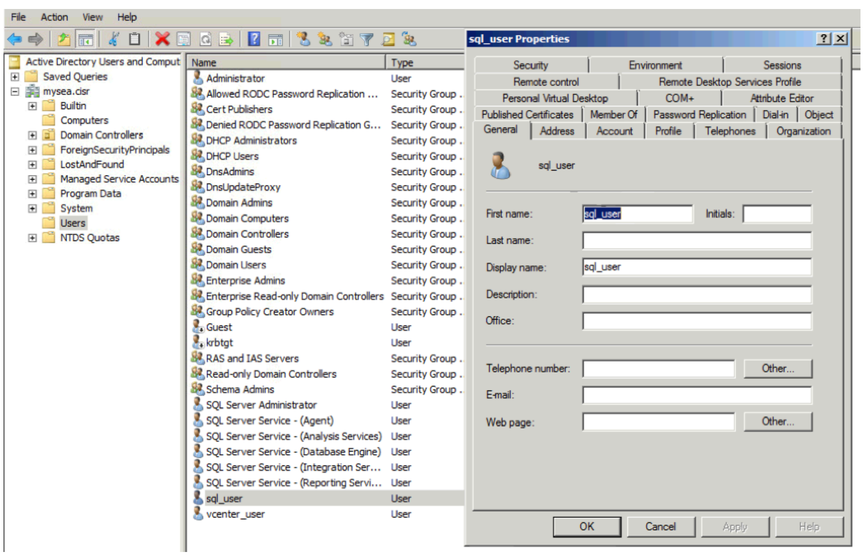


Figure 27. Active Directory Users and Computers.

   1.4. Log out of *CISR_DC01*.
2. Create a new VM (see Appendix C).
3. Map SQL Server 2008 ISO to the VM (see Appendix L, sec. C).
4. Log on to the VM just created (Microsoft SQL Server 2008 R2).
5. Navigate to the CD-ROM drive and launch the SQL Server 2008 R2 installation file.
6. Start the install wizard; the wizard will validate all prerequisites for installation.

7. Run the **System Configuration Checker** (see Figure 28). Resolve any errors reported by the **Setup Support Rules** screen (see Figure 29) before proceeding.

Figure 28. SQL Server Installation Center.

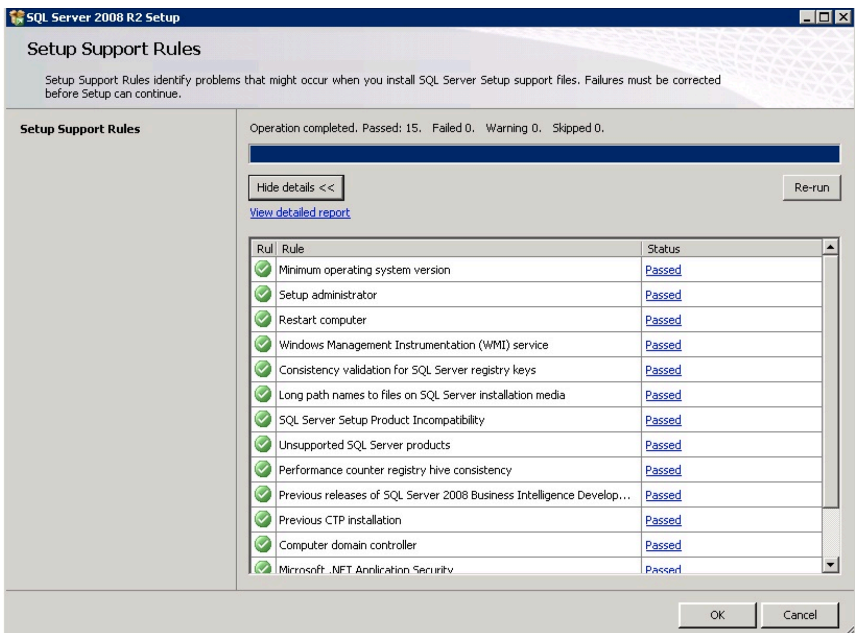


Figure 29. Setup Support Rules.

8. Follow the procedures from the Microsoft TechNet instructions[7].
   8.1. On the **Feature Selection** page, select the features and specify the directory location (see Figure 30).

[7] "How to: Install SQL Server 2008 R2 (Setup)," available online at: http://technet.microsoft.com/en-us/library/ms143219(v=sql.105).aspx.

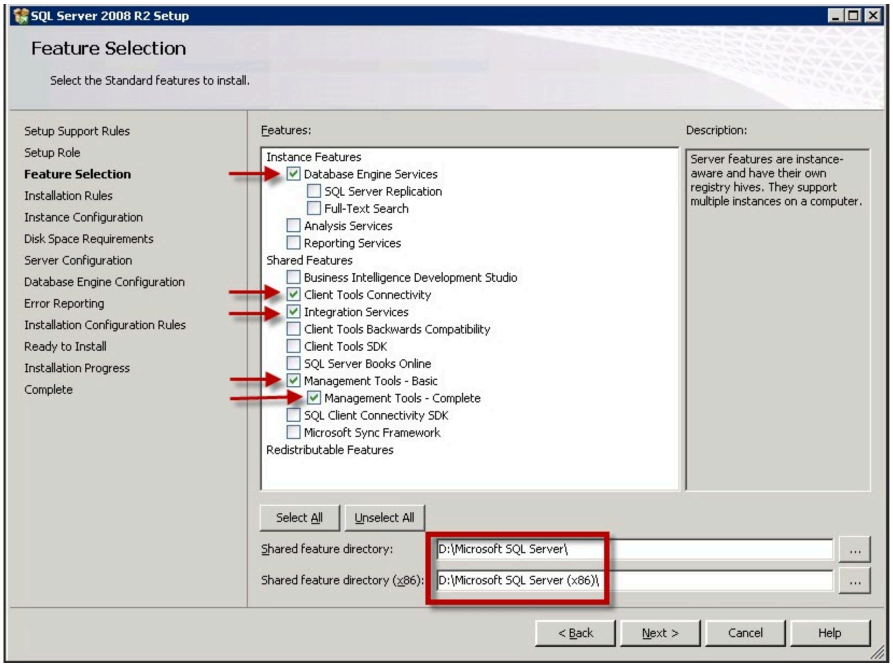


Figure 30. Feature Selection Page.

8.2. On the **Instance Configuration** page, choose the **Default Instance Option**. Use the default instance ID (`MSSQLSERVER`) and default **Instance Root Directory** (`D:\Microsoft SQL Server\`) shown in Figure 31.

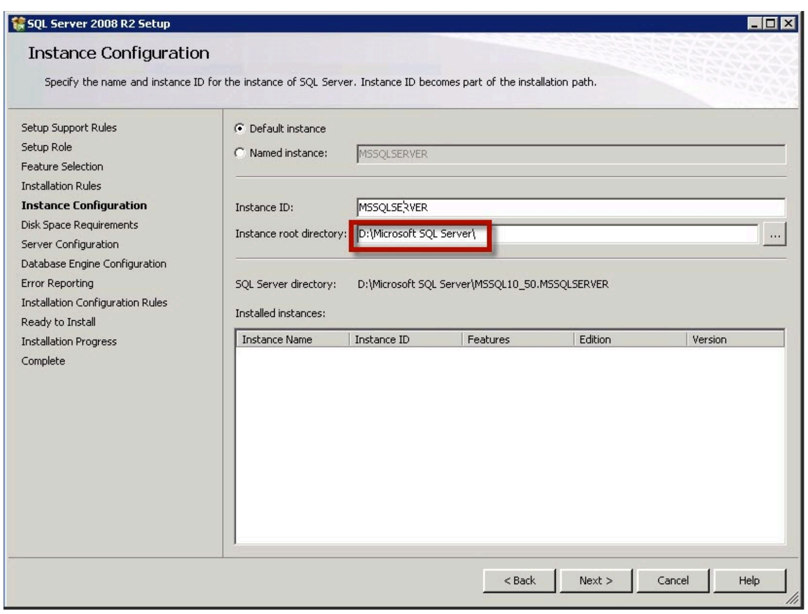


Figure 31. Instance Configuration Page.

8.3. On the **Server Configuration** page, click the **Use the same account for all SQL Server services** button. Enter the password into the Password field (see Figure 32).

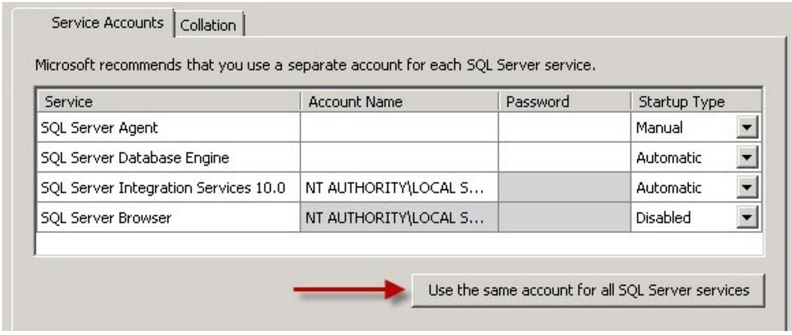


Figure 32. Server Configuration page – Service Accounts Tab.

8.4. On the **Server Configuration** page, change the "SQL Server Agent" entry to have "Startup Type" as **Automatic**.

8.5. On the **Database Engine Configuration – Account Provisioning** page, choose **Mixed Mode Authentication** for "Security Mode." Then, enter the credentials for the SQL system `Administrator` account (from Step 1).

9. Install any Windows Updates.

10. Reboot.

B. CONFIGURE MICROSOFT SQL SERVER DATABASE(S)

1. Launch the "Microsoft SQL Server Management Studio" application, using the **Start** menu.
2. Enter logon credentials, when prompted (see Appendix A, Table 8).
3. Within the **Object Explorer** panel, right click on **Databases** and select **New Database** (see Figure 33).

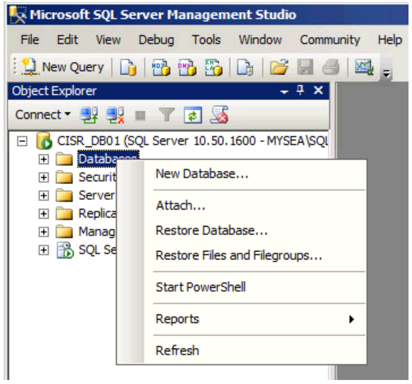


Figure 33. New Database Selection Menu via Database Folder.

4. From the **General** page, choose the new database name “vCenter_DB” (see Figure 34).

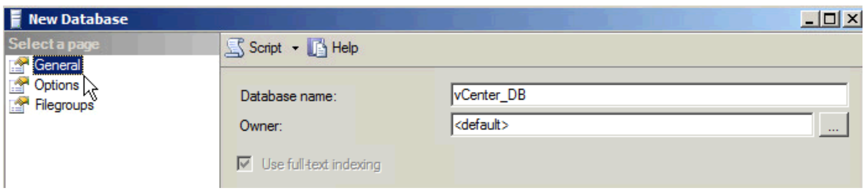


Figure 34. New Database “General” Settings Page.

5. From the **Options** page, change the “Recovery model” to **Simple**. Now click **OK** and the database will be created (see Figure 35).

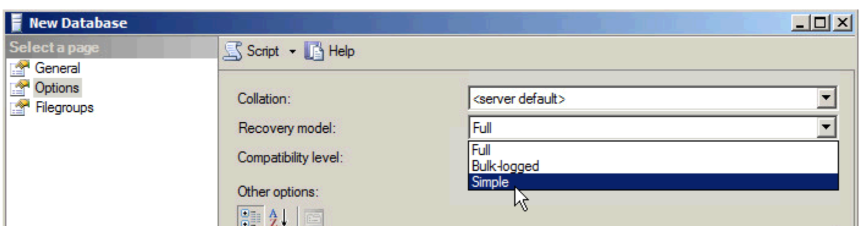


Figure 35. New Database “Options” Settings Page.

6. Follow the below steps, to create a dedicated vCenter user. This user will be used by vCenter to connect to the SQL Server database.
   6.1. From the **Object Explorer** panel, expand the **Security** folder, then right-click the **Logins** folder to select **New Login** (see Figure 36).

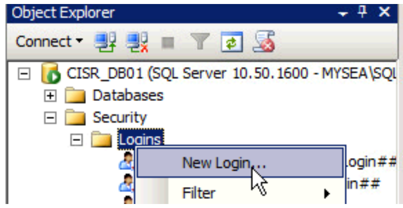


Figure 36. New Login Menu Option.

6.2. From the **General** page, choose a username (`vcenter_user`) and password for the new vCenter user. Set the "Default Database" to **vCenter_DB** (previously created in Step 4). *Don't click "OK" yet* (see Figure 37).

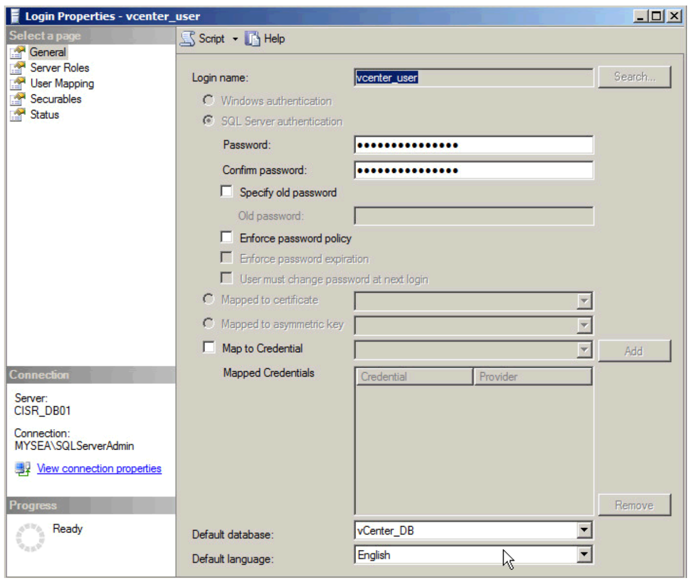


Figure 37. General Page for `vcenter_user`.

6.3. Select the **User Mapping** page from the left panel, then map the **`vcenter_user`** to the **msdb** and **vCenter_DB** databases. For both entries, set the "Default Schema" to **dbo**. Select the **db_owner** checkbox for both databases. Click **OK** (see Figure 38).

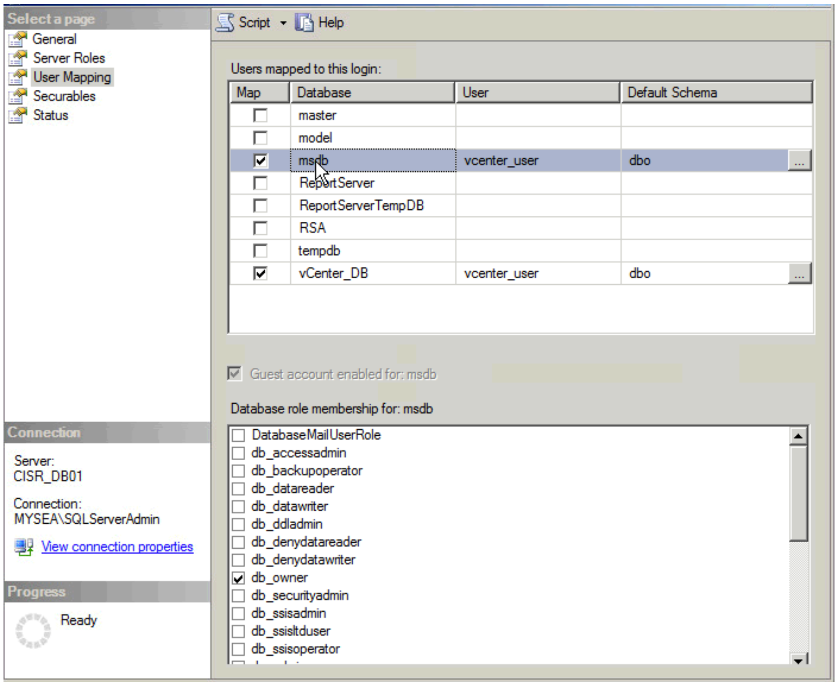


Figure 38. User Mapping Page for vcenter_user.

7. Add a new ODBC Data Source Name (DSN) to the system, following the steps below:
   7.1. From the Start Menu, select **Administrative Tools>Data Sources** (ODBC).
   7.2. Select the **System DSN** tab and click **Add**.
   7.3. Select SQL Server Native Client 10.0, then click the **Finish** button (Figure 39).

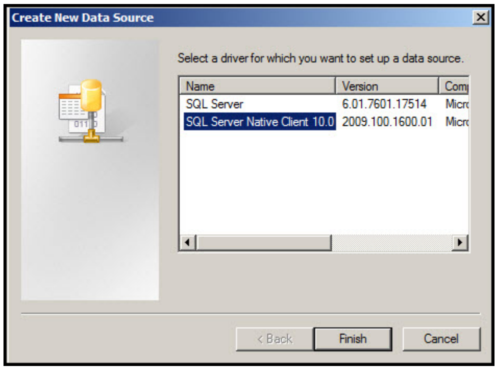


Figure 39. Data Source Driver Selection Page.

8. The **Create a New Data Source to SQL Server** window will appear. Enter the name **vCenter_DB** for the DSN and **CISR_DB01** for the DNS name of the SQL Server (Figure 40).

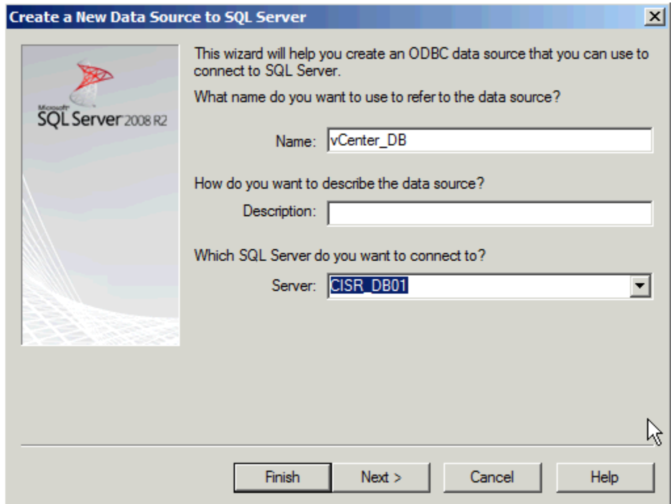


Figure 40. New Data Source to SQL Server Wizard.

9. Enter the user credentials (from Step 6.2). Then, click **Next**.
10. Click **Next** again (no changes needed on this page)
11. Click **Next** again (no changes needed on this page)
12. Review the **SQL Server Native Client** summary page, see Figure 41. If satisfied, click **Test Data Source** to test the connection to the SQL Server. If the connection is successful, then "TESTS COMPLETED SUCCESSFULLY" should appear on the next page (see Figure 41).

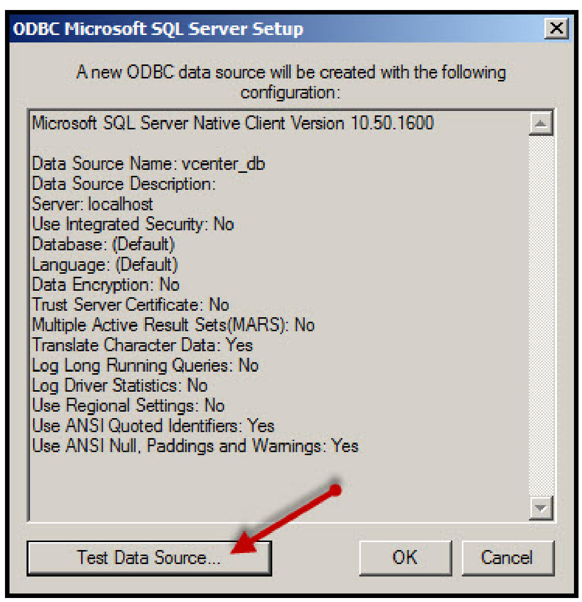


Figure 41. ODBC Data Source Summary Page.

13. Proceed to vCenter Installation.

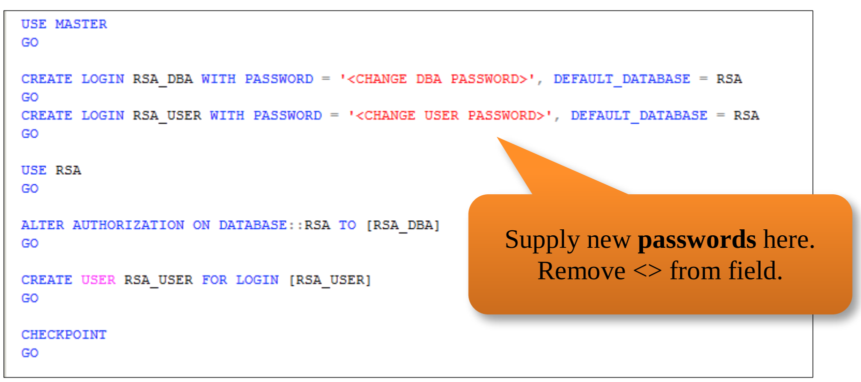
C. PREPARE SINGLE SIGN ON DATABASE.

Since we install vCenter using an existing database, the installer will prompt for the usernames/passwords for a database administrator (`RSA_DBA`) and a database user (`RSA_USER`), as part of installing the vCenter Single Sign-On component. The users are created manually using a SQL script, found on the installation media. The following instructions explain how to use this script.

1. On the SQL Server, navigate to the following location on the vCenter installation media:

   `\Single Sign On\DBScripts\SSOServer\schema\mssql`

2. Double click on the following script: `rsaIMSLiteMSSQLSetupUsers.sql`. The Microsoft SQL Server Management Studio will launch.
3. Supply new passwords for the `RSA_DBA` and `RSA_USER` (see Figure 42).



```
USE MASTER
GO

CREATE LOGIN RSA_DBA WITH PASSWORD = '<CHANGE DBA PASSWORD>', DEFAULT_DATABASE = RSA
GO
CREATE LOGIN RSA_USER WITH PASSWORD = '<CHANGE USER PASSWORD>', DEFAULT_DATABASE = RSA
GO

USE RSA
GO

ALTER AUTHORIZATION ON DATABASE::RSA TO [RSA_DBA]
GO

CREATE USER RSA_USER FOR LOGIN [RSA_USER]
GO

CHECKPOINT
GO
```


Figure 42. rsaIMSLiteMSSQLSetupUsers.sql Script.

4. Run the script by clicking **Execute**.

This script: (1) creates the `RSA_DBA` and `RSA_USER` login accounts, and (2) creates two database users (`dbo` and `RSA_USER`) for the RSA database. The `RSA_USER` is mapped to the `RSA_USER` login account; the `dbo` user is mapped to the `RSA_DBA` login (see Figure 43).

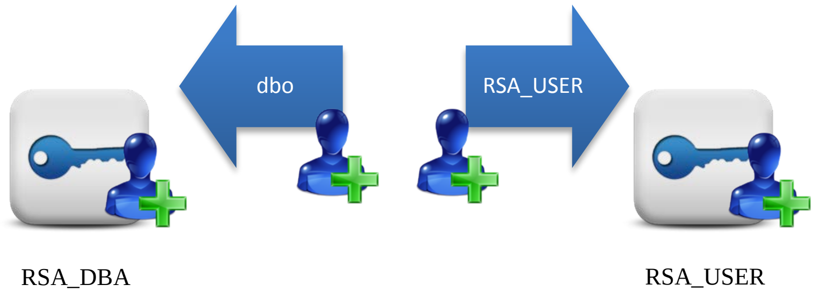


Figure 43. RSA_DBA and RSA_USER Mapping.

APPENDIX G. INSTALLING VCENTER

Following both Appendix B and Appendix C, all prerequisite steps must be satisfied to install vCenter 5.1. Prior to installation, all prerequisite, pre-installation tasks (see Appendix A and Appendix B) should be satisfied as described in the vSphere Guide.[8] The vCenter Server software itself is installed in a matter of minutes using the “vCenter Server Simple Install” option through the VMware vCenter Installer. There are two options available during the install, “Simple Install” or individual component install. We used the Simple Install option (see Figure 44) for the test bed, which sequentially installs the following components in the required installation sequence on the same host or VM: vCenter Single Sign-On, Inventory Service, and vCenter.

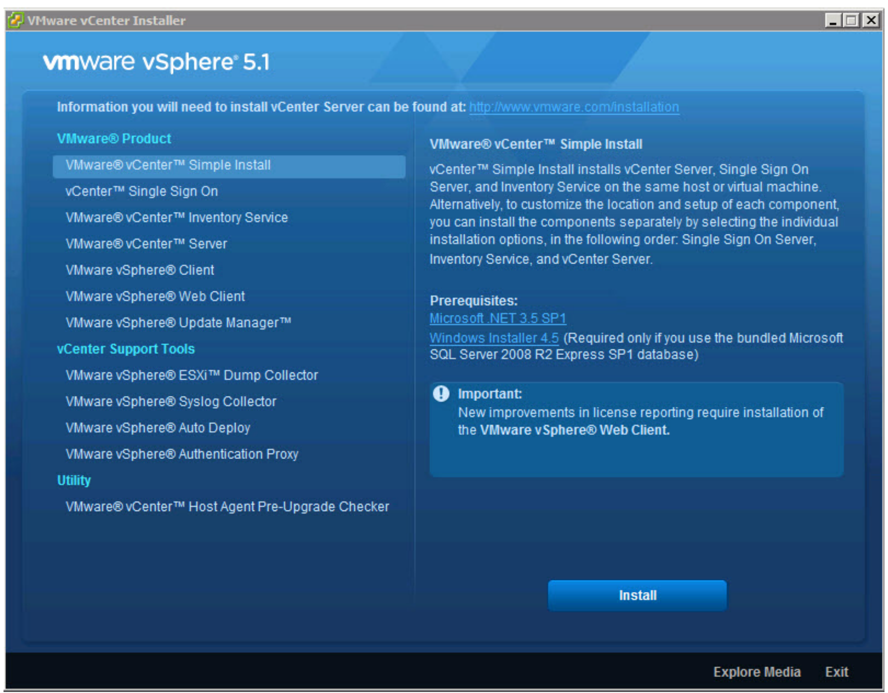


Figure 44. VMware vCenter Installer.

[8] vSphere Installation and Setup Guide, Chapter 10, page 221, available online at: http://pubs.vmware.com/vsphere-51/topic/com.vmware.ICbase/PDF/vsphere-esxi-vcenter-server-51-installation-setup-guide.pdf.

Refer to VMware's vSphere Installation and Setup Guide for the Simple Install instructions[9]. After the Simple Install, re-launch the vCenter Installer and install the vSphere Client. Follow the self-guided installation wizard to complete the installation.

[9] vSphere Installation and Setup Guide, Chapter 11, pages 247-250 available online at: http://pubs.vmware.com/vsphere-51/topic/com.vmware.ICbase/PDF/vsphere-esxi-vcenter-server-51-installation-setup-guide.pdf.

APPENDIX H. CREATING VMWARE NETWORK INFRASTRUCTURE

Prior to deploying Serengeti virtual appliance, we had to prepare the VMware layer to comply with all the requirements listed in section 2.4 of the VMware Serengeti User's Guide. After the installation of the vCenter we decided to organize the three production ESXi hosts in to a single cluster in order to enable CPU and RAM sharing. This was necessary in order to minimize provisioning time for the Hadoop cluster nodes. Another prerequisite that we had to meet was the creation of a VMware resource pool. The resource pool requires High Availability (HA) and Distributed Resource Scheduler (DRS) options to be running on the vCenter cluster. HA does not provide 100% availability of VMs, but rather provides higher availability by rapidly recovering VMs on failed hosts and VMware DRS is a load balancing utility that assigns and moves computing workloads to available hardware resources in a virtualized environment. Finally, we had to setup the internal layer-two infrastructures on each host, by creating and configuring a number of virtual switches and virtual interfaces.

A. CONFIGURE VMWARE LAYER

1. Log on **VADMIN1** server as an `Administrator` (see Appendix A, Table 8).
2. Double click on **VMware vSphere Client** icon on the Desktop (see Figure 45) and login vCenter server using `Administrator` credentials (see Appendix A, Table 8).

Figure 45. VMware vSphere Client Login Screen.

3. From the left pane, right-click **CISRVCNTR.mysea.cisr**, select **New Datacenter** (see Figure 46). Enter `CISR` as the name of the new Datacenter.

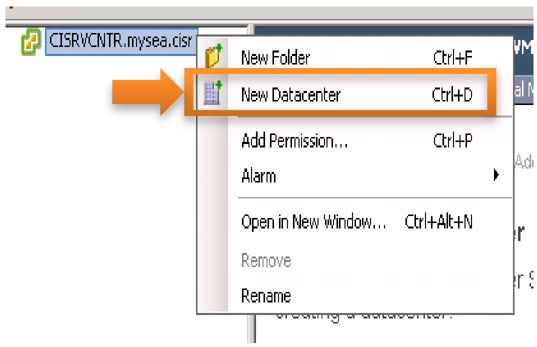


Figure 46. New Datacenter Menu Selection via vSphere Client.

4. Click on the newly create Datacenter, select the **IP Pools** tab from the left pane and click on **Add**. Name the New IP Pool as `CISR`, click on the **DHCP** tab, put a check next to **IPv4 DHCP Present** then click **OK** (see Figure 47).

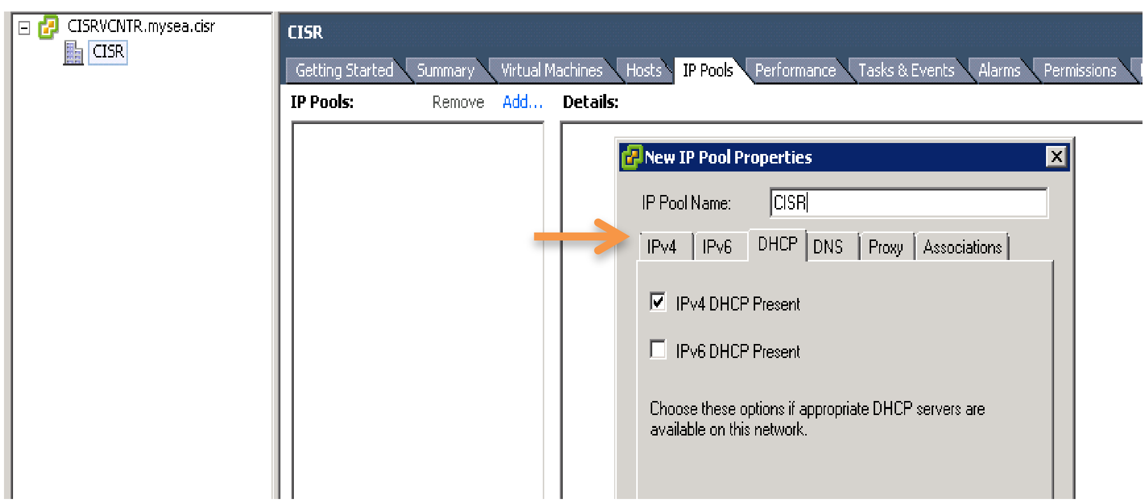


Figure 47. New IP Pool Properties for New Datacenter.

5. Right-click on the newly create Datacenter, select **New Cluster** (see Figure 48) and name `Serengeti` (see Figure 49). Click on **Next**, and continue to do so until you finally click on **Finish**, while accepting all default options.

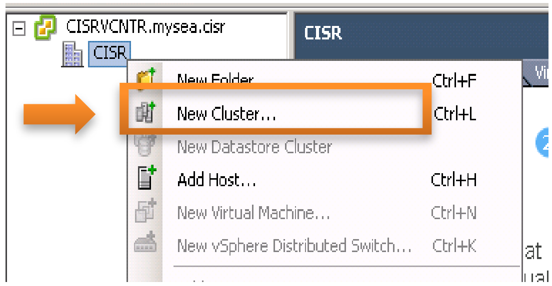


Figure 48. New Cluster Menu Selection via CISR Datacenter.

Note: If you want to deploy multiple hosts per cluster, ensure that **Turn On vSphere HA** and **Turn On vSphere DRS** are checked (see Figure 49).

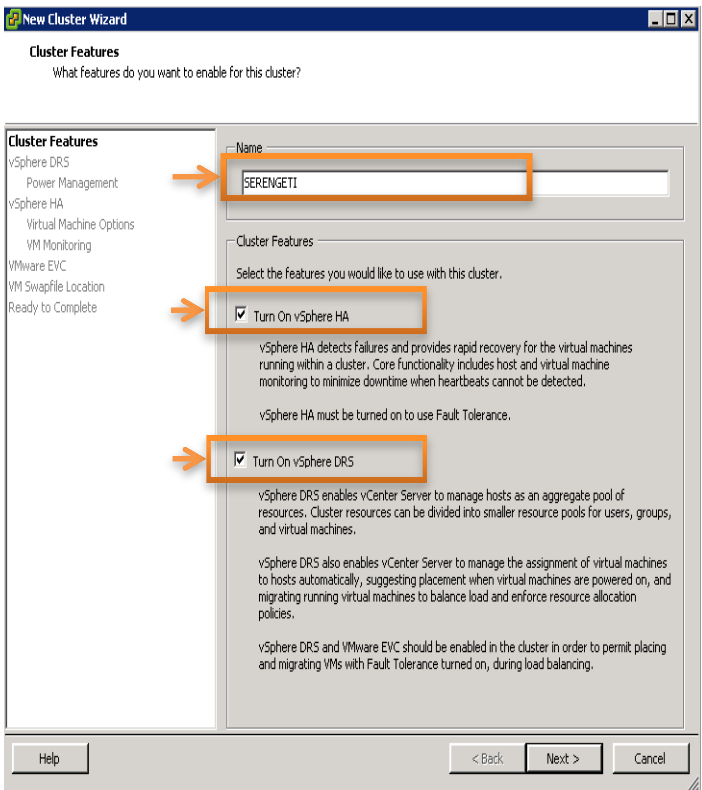


Figure 49. New Cluster Wizard.

6. Right-click on the newly created cluster and select **Add Host**. Enter the IP address of **R1S1** (see Appendix A, Table 9) and required login credentials (see Figure 50). Click **Yes** on the Security Alert pop-up (see Figure 50), and then click **Next** twice, enter **VMware license** acquired previously, and continue clicking **Next** until you click **Finish**, while accepting all default settings.

Note: We added *R1S2* and *R1S3* to the cluster, in order to reduce provisioning time.

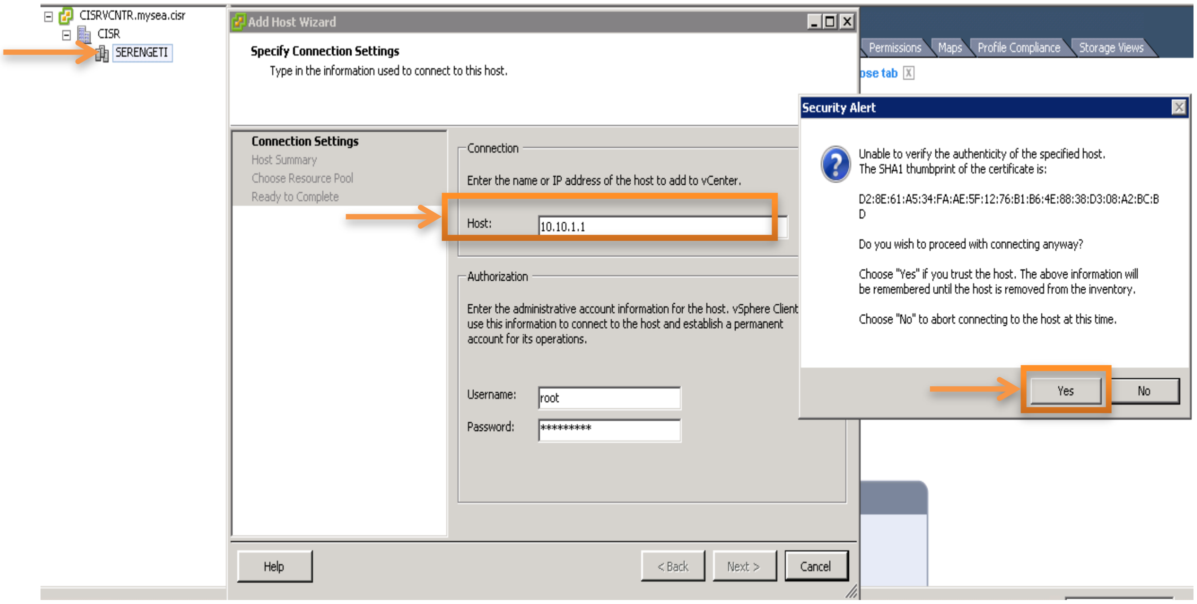


Figure 50. Add Host Wizard.

7. Right-click the cluster and select **New Resource Pool** (see Figure 51). Name the Resource Pool SERENGETI then accept all defaults and click **OK**.

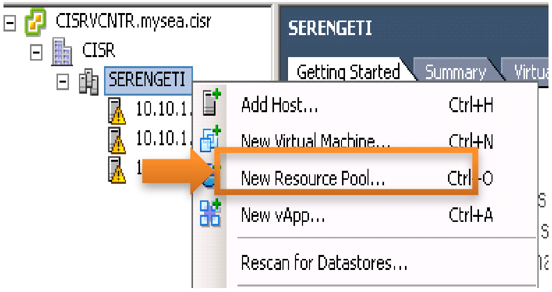


Figure 51. New Resource Pool Menu via Serengeti Cluster.

B. CONFIGURE VIRTUAL NETWORK LAYER

1. Highlight the first host, then in the right pane click on **Configuration** tab. In the Hardware section, click on **Networking** (see Figure 52).

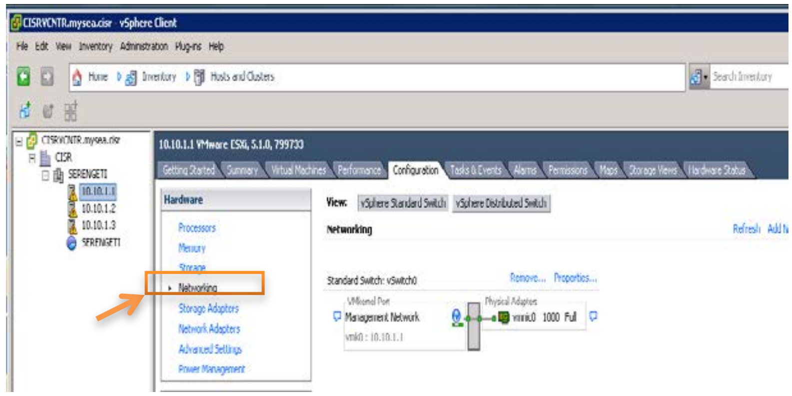


Figure 52. Host Hardware and Network Configuration Settings.

2. Click on the **Properties** for vSwitch0, highlight vSwitch and click on **Add** (see Figure 53). Click **Next,** enter network label of your choosing (see Figure 54), click **Next** and then click **Finish.**

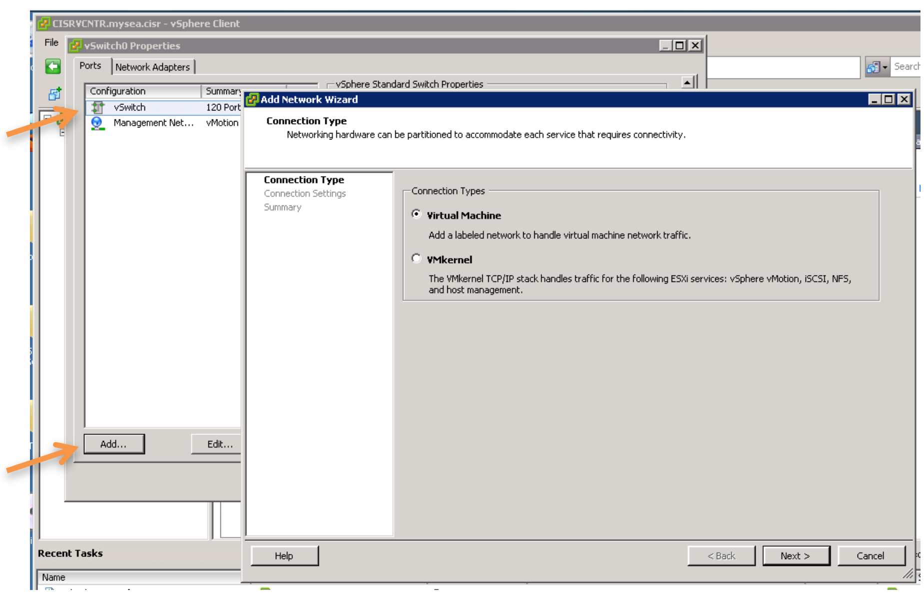


Figure 53. Add Network Wizard – Connection Type Selection.

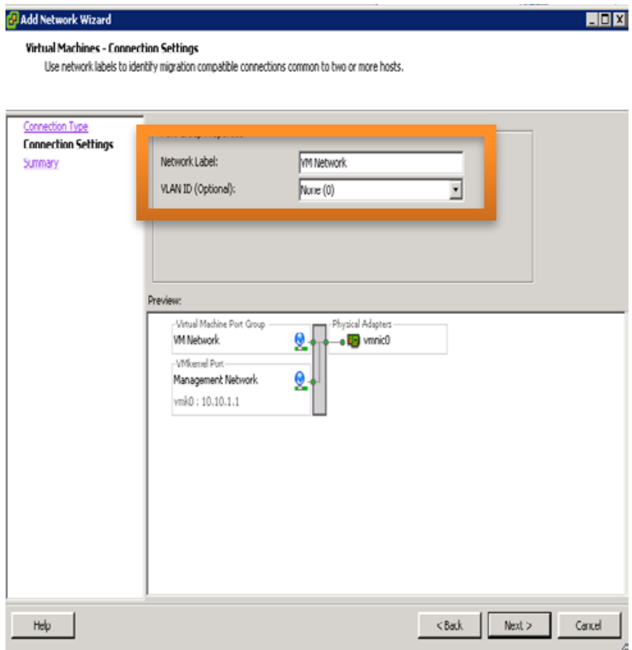


Figure 54. Add Network Wizard – Connection Settings.

3. From the Software section, select **DNS and Routing** (see Figure 55) then click on **Properties**.

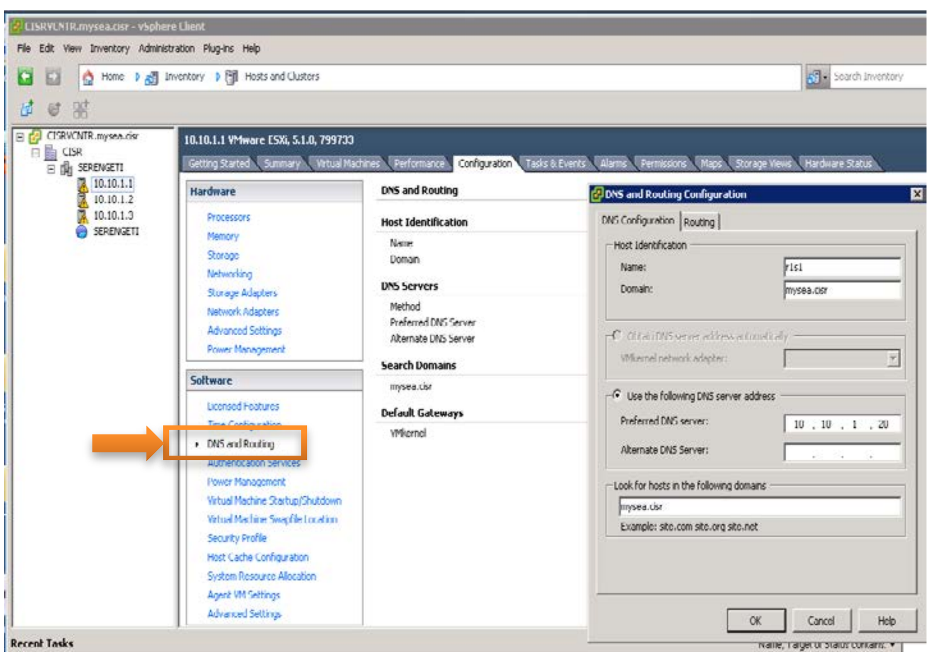


Figure 55. DNS and Routing Configuration.

4. **Enter Name:** R1S1 (naming conventions in Appendix A, Table 9), **Domain:** "mysea.cisr." Select **Use the following DNS server address,** enter the IP address of the internal DNS server (see Appendix A, Table 9).
5. Click on the **Routing** tab (see Figure 56) and ensure that **Default Gateway** is configured with the respective host's IP address (see Appendix A) then click **OK**.

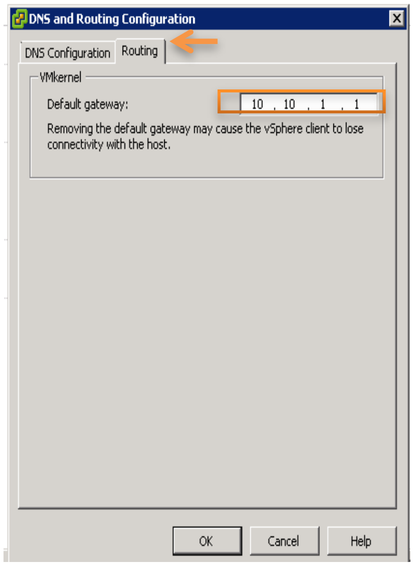


Figure 56. DNS and Routing Configuration.

Note: Perform NTP server configuration (see Appendix E) prior to performing steps 7 thru 10 of this section.

6. Select **Time Configuration** from the Software pane, and then click **Properties** in the **Configuration** tab (see Figure 57).
7. Check **NTP Client Enabled** option and click on **Options** button (see Figure 57).
8. Highlight **NTP Settings** and click **Add**. Type the IP address of Active Directory Domain Controller (see Appendix A, Table 9) and click **OK**.
9. Ensure that **Restart NTP service to apply changes** is checked, and click **OK**. (see Figure 57).

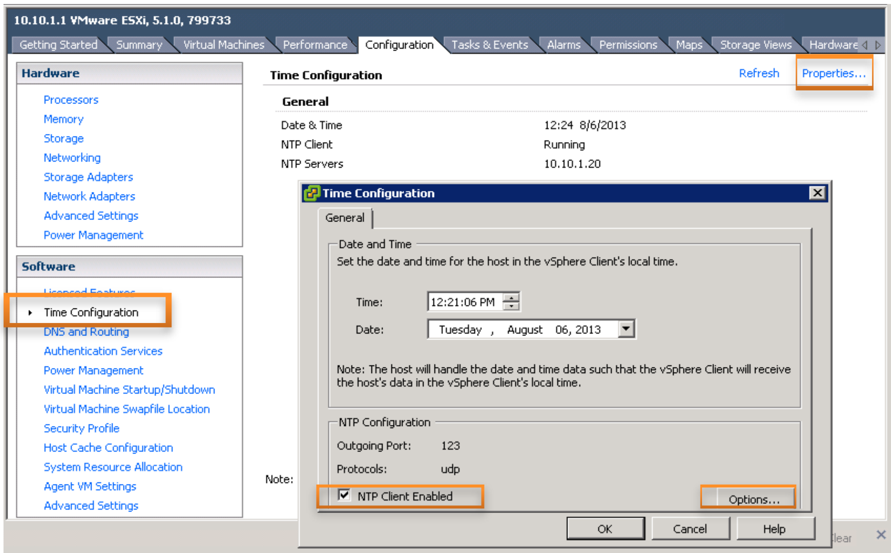


Figure 57. Time Configuration Settings.

10. Repeat steps **1** thru **10** of this section for all ESXi hosts.

11. Double click on **VMware vSphere Client** icon on the Desktop and login to **R3S1** using `Administrator` credentials (see Appendix A, Table 8).

12. Highlight the host (10.10.1.X) again then in the right pane click on the **Configure** tab. In the Hardware section select **Networking** (see Figure 58).

13. Click on Add > Networking > Next.

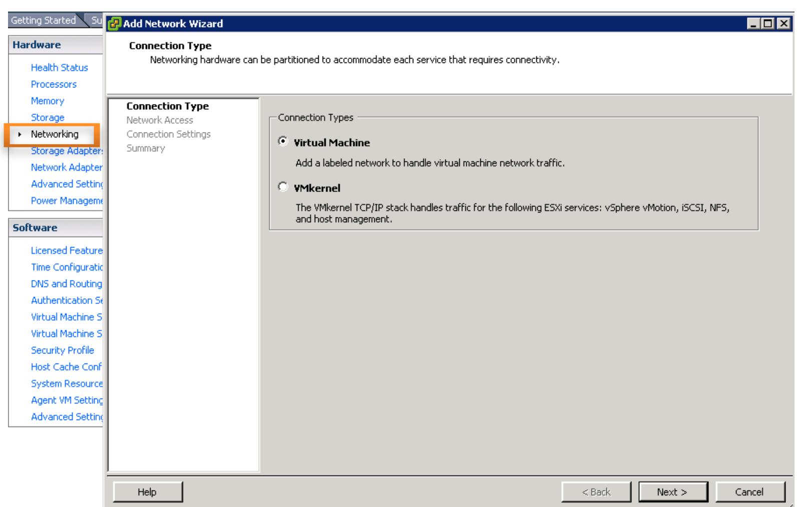


Figure 58. Add Network Wizard – Connection Type Setting.

14. Ensure **vmnic1** is checked (see Figure 59) and click **Next**.

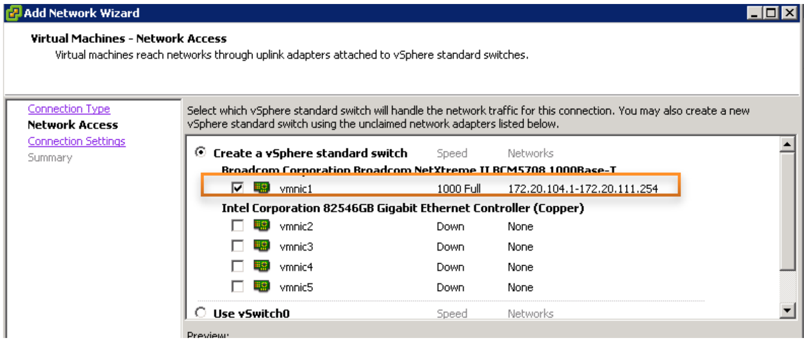


Figure 59. Add Network Wizard – VM Network Access Configuration Selection.

15. Change the network label to **CAMPUS NETWORK** (see Figure 60) then click on **Next** and **Finish**.

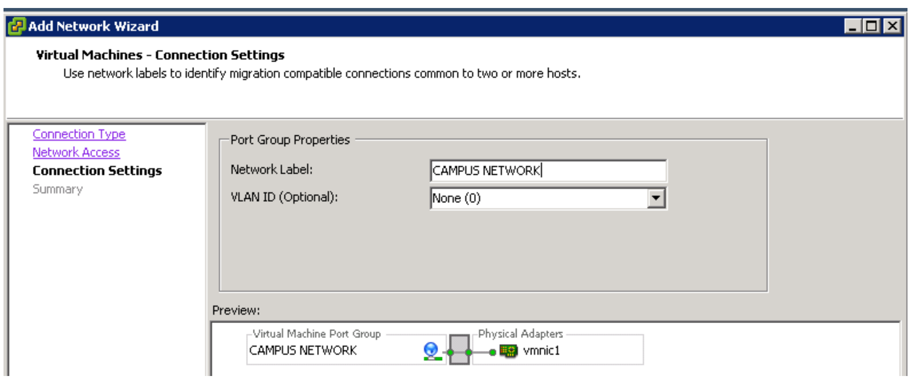


Figure 60. Add Network Wizard – VM Connection Settings.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX I. CREATING DEFAULT SERENGETI MANAGEMENT SERVER

Serengeti version 0.8 was used because it was the most recent version available at the time we started the project. Serengeti 0.8 is an OVA and must be downloaded from the VMware website. A VMware account is required to download software from the site. Registration is free and can be done from the VMware website at: http://www.vmware.com.

A. DOWNLOADING SERENGETI:

1. Once the account registration is completed, browse to the VMware home page at the link above.
2. Click on **Support and Downloads** then select **All Downloads** from the Product Downloads column (see Figure 61).

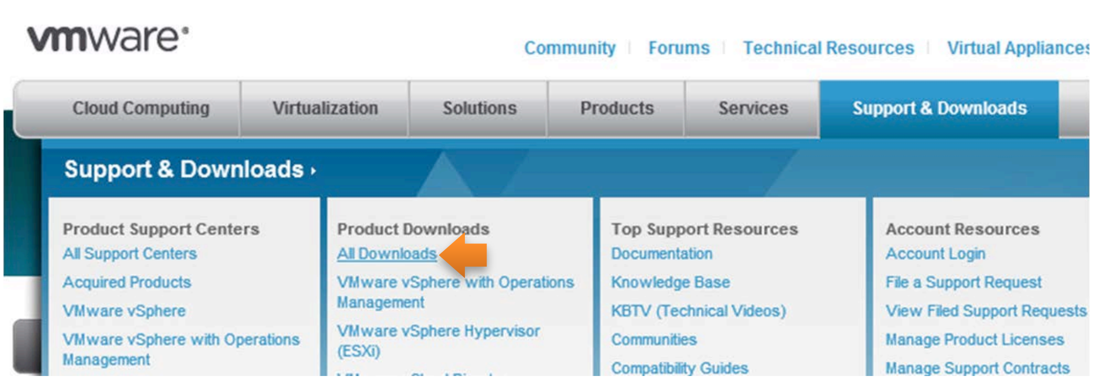


Figure 61. VMware Support & Downloads Webpage.

3. Click on the **Products A-Z** tab, then drag the mouse over **Serengeti** and click **View Components** (see Figure 62).

Figure 62. VMware Products A-Z Page.

4. Click **Go to Downloads** button (see Figure 63).

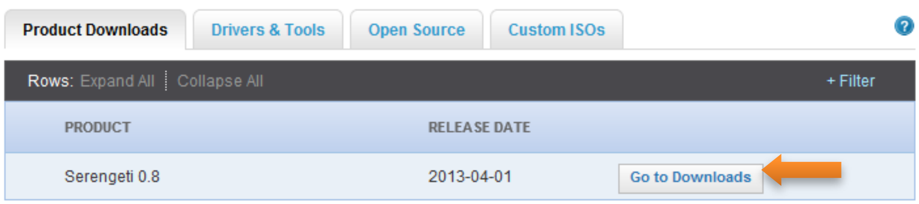


Figure 63. Serengeti 0.8 Product Page.

5. Click the **Download** button (see Figure 64).

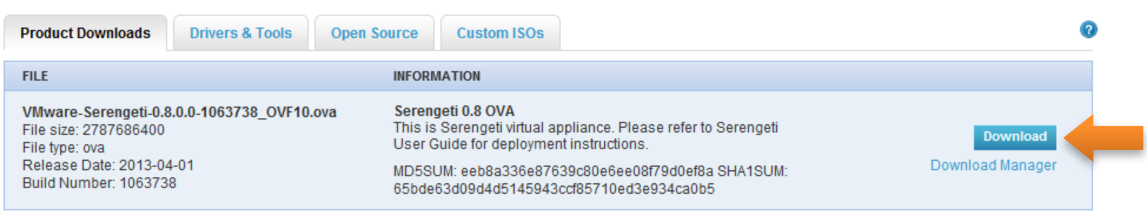


Figure 64. Serengeti 0.8 Download Page.

6. If not already logged in, enter VMware account credentials when prompted and click **Log In** (see Figure 65).

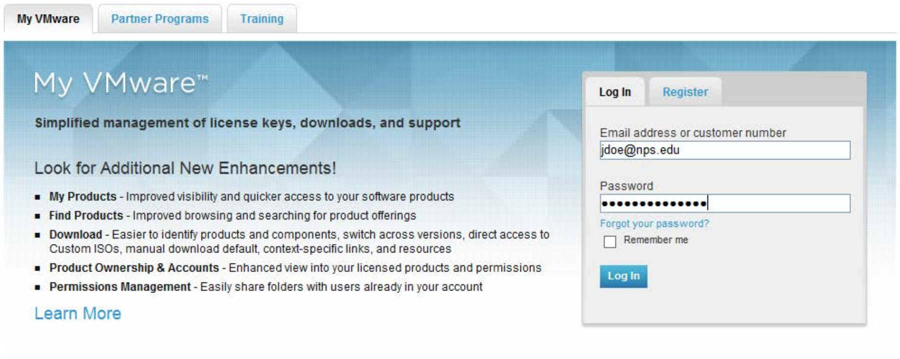


Figure 65. My VMware Login Page.

7. Agree to the terms and conditions of VMware's license agreement and click **Accept.**
8. Select **Save As**, navigate to the desired destination directory, and then click **Save** (see Figure 66). Serengeti is distributed as an open virtualization format (OVF) package.

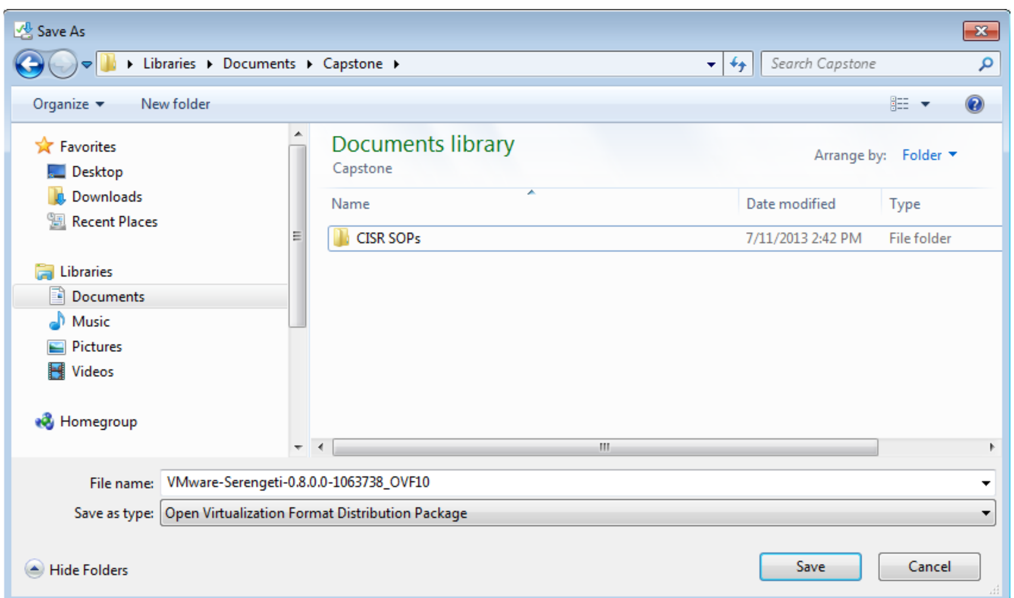


Figure 66. Saving Serengeti OVF in Destination Directory.

B. INSTALL SERENGETI

Installation of Serengeti requires the vCenter server to be properly installed, configured, and running in accordance with Appendix G.

1. Log into vCenter via the vSphere Client.
2. Click **File** and select **Deploy OVF Template** (see Figure 67).

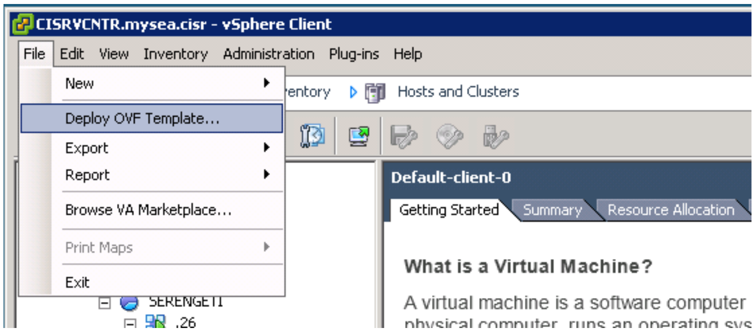


Figure 67. Deploy OVF Template Menu Action.

3. Click **Browse** and navigate to the directory where the Serengeti OVF is saved. Select the OVF file and click **Open** (see Figure 68).

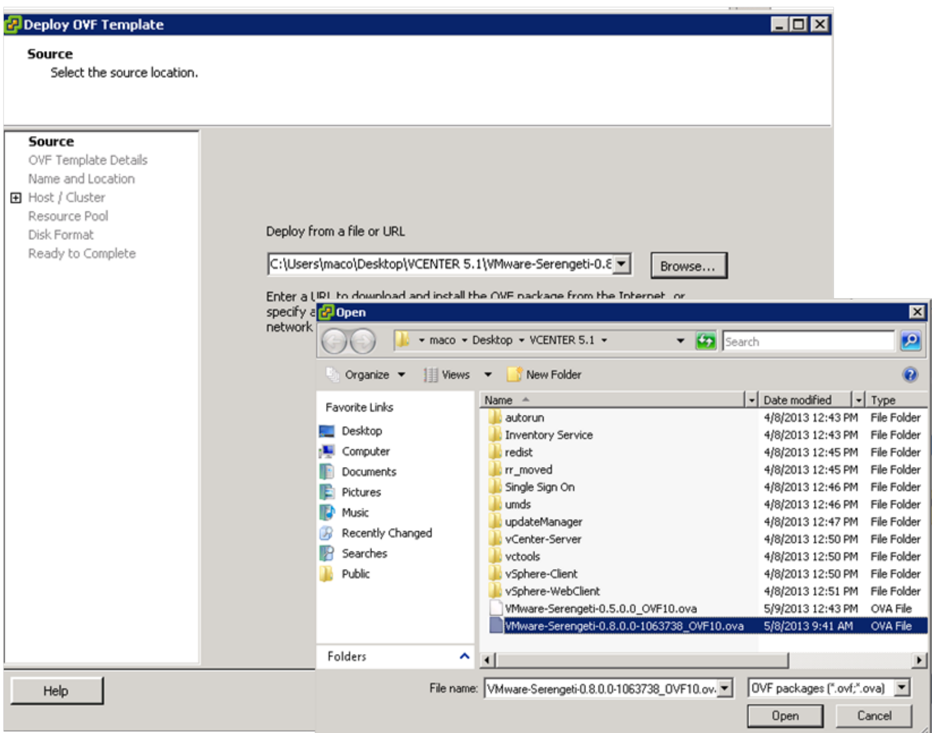


Figure 68. Source Selection for OVF File.

4. Click **Next** to confirm the source location.
5. Click **Next** to acknowledge OVF template details.
6. Click **Accept** to accept the license agreement and click **Next.**

7. Specify the Serengeti virtual appliance name (we used Serengeti-Test) and select the CISR Center (see Appendix H) in which to install Serengeti, and then click **Next** (see Figure 69).

> **Note**: Only alphanumeric letters ("a-z." "A-Z"), numbers ("0–9"), hyphen ("-"), and underscore ("_") can be used when naming a virtual appliance.

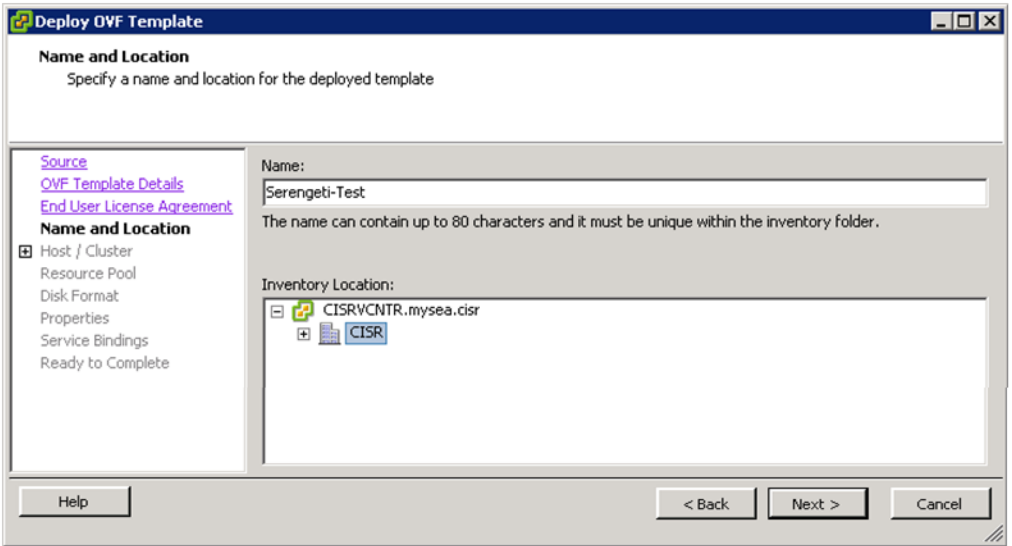


Figure 69. Name and Location Specification Page for OVF.

8. Select the SERENGETI cluster to install Serengeti (see Figure 70), and then click **Next.**

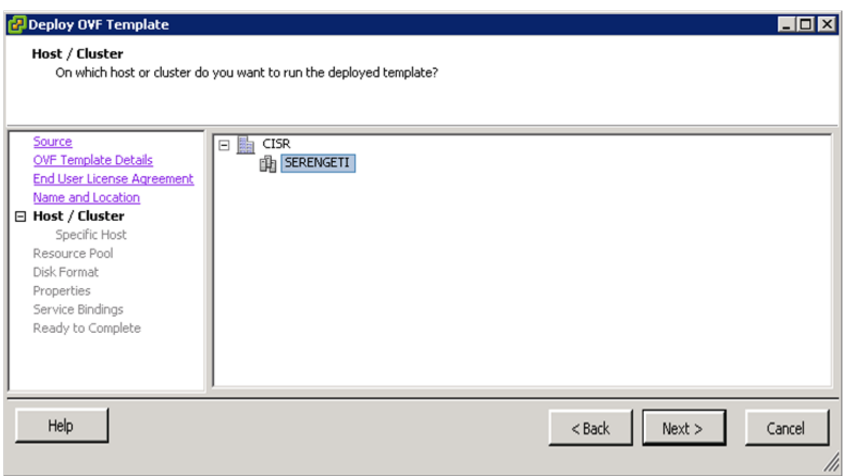


Figure 70. Host / Cluster Deployment Selection Page.

9. Select the resource pool on which to deploy the template (see Figure 71) and then click **Next.**

Note: You must deploy Serengeti in a top-level resource pool.

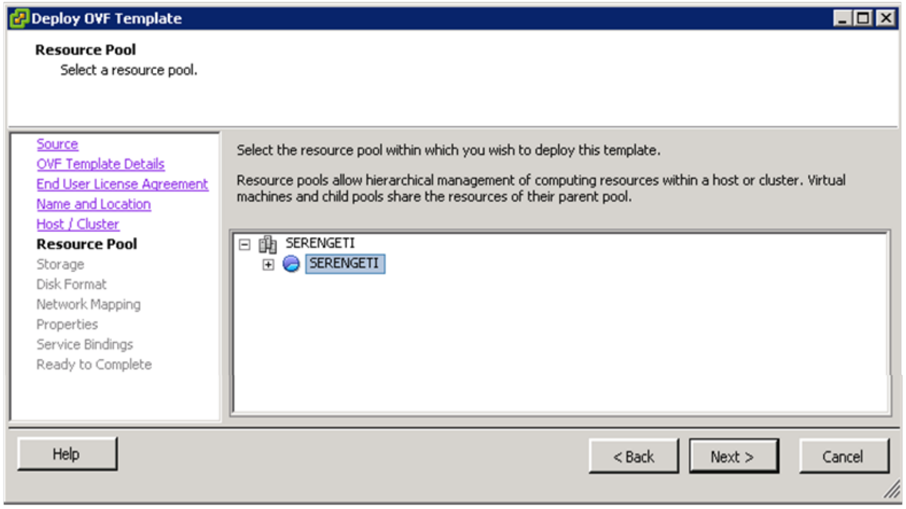


Figure 71. OVF Resource Pool Selection Page.

10. Select a datastore which to install Serengeti, and then click **Next** (see Figure 72).

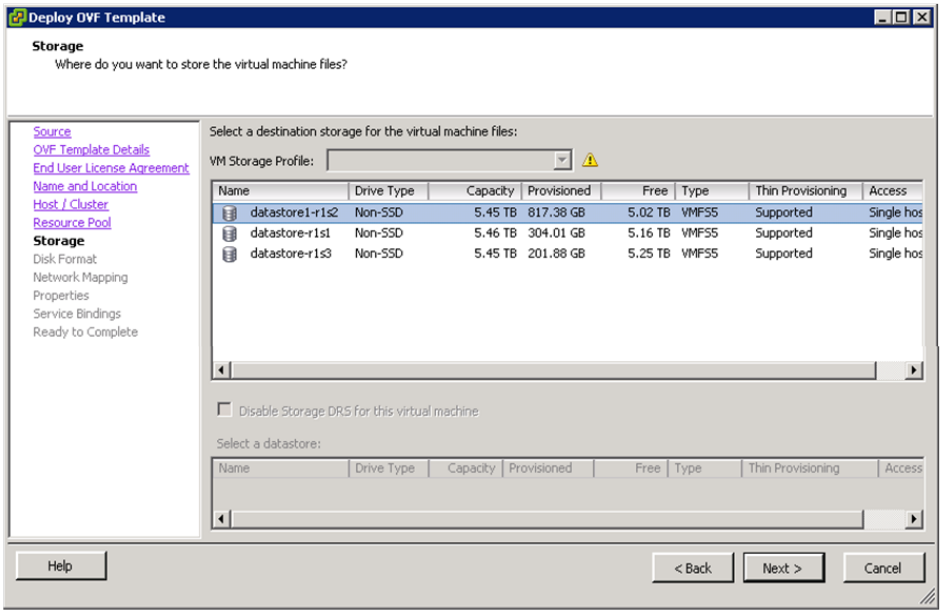


Figure 72. OVF Storage Designation Page.

11. Select **Thin Provision** format for the virtual disks (see Figure 73), and then click **Next.**

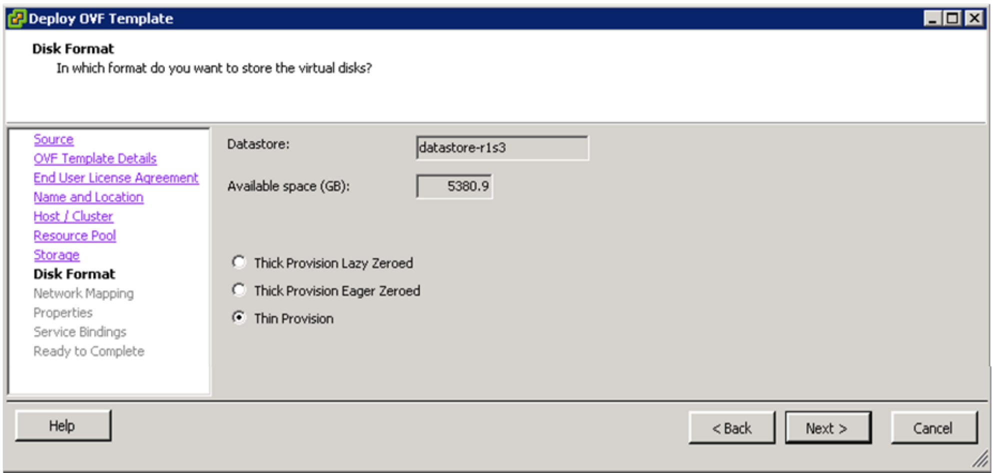


Figure 73. OVF Disk Format Selection Page.

12. Select the destination network that will allow Serengeti to communicate with the vCenter server (see Figure 74), and then click **Next**.

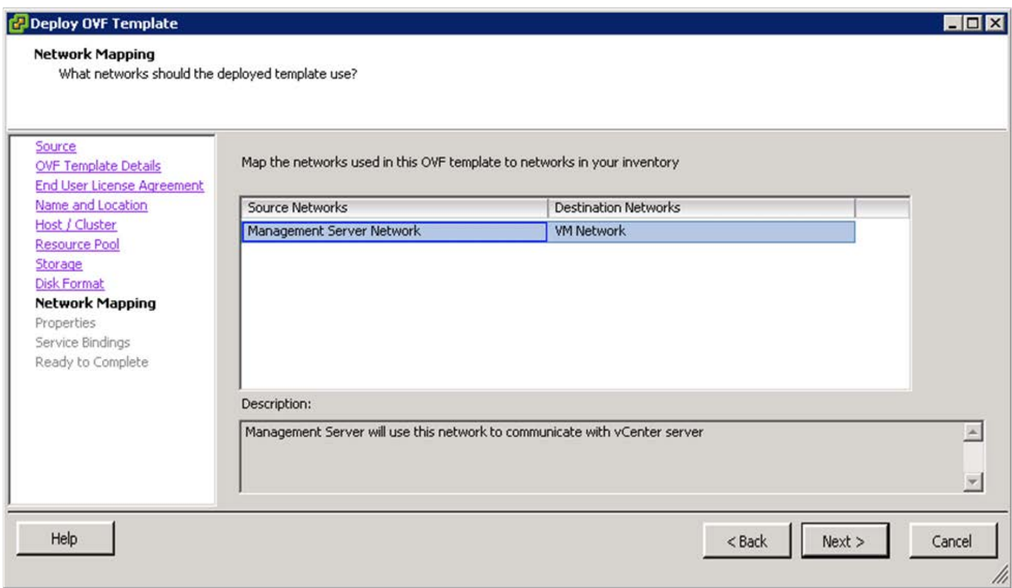


Figure 74. OVF Network Mapping Page.

13. Set the properties for the Serengeti deployment (see Figure 75), and then click **Next**.

Note: If using a static IP address, ensure the selected IP address is in the same subnet as vCenter server (see Appendix A, Table 9).

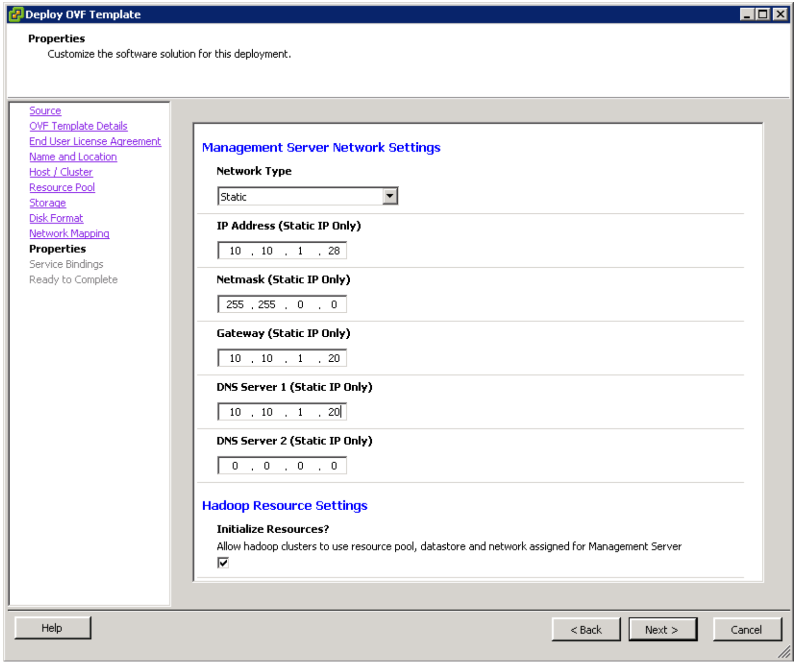


Figure 75. OVF Management Server Network Settings Page.

14. Verify the default binding to the **vCenter** Extension Service (se Figure 76), and then click **Next.**

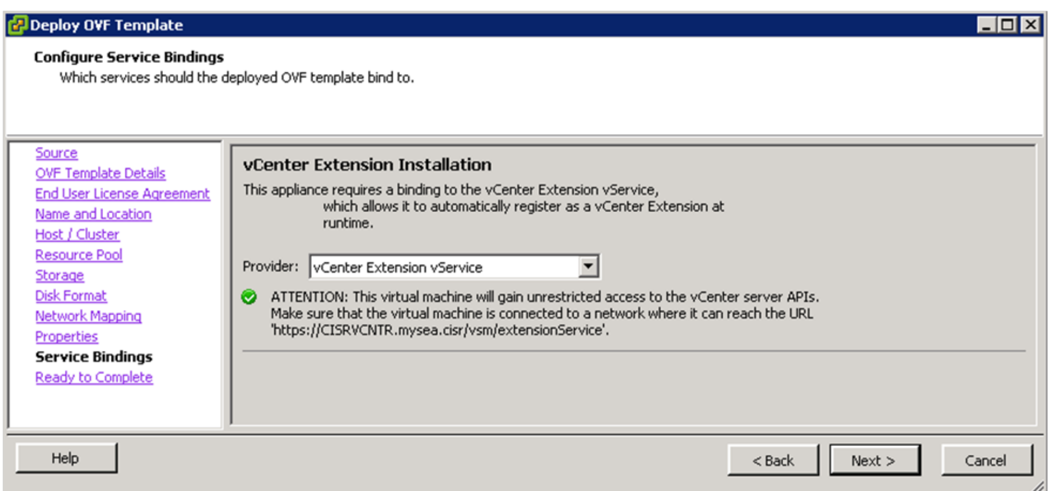


Figure 76. OVF vCenter Extension Installation Page.

15. Verify the options listed are correct. Use the Back button to make changes if necessary. If desired, check **Power on after deployment**, and then click **Finish** (see Figure 77).

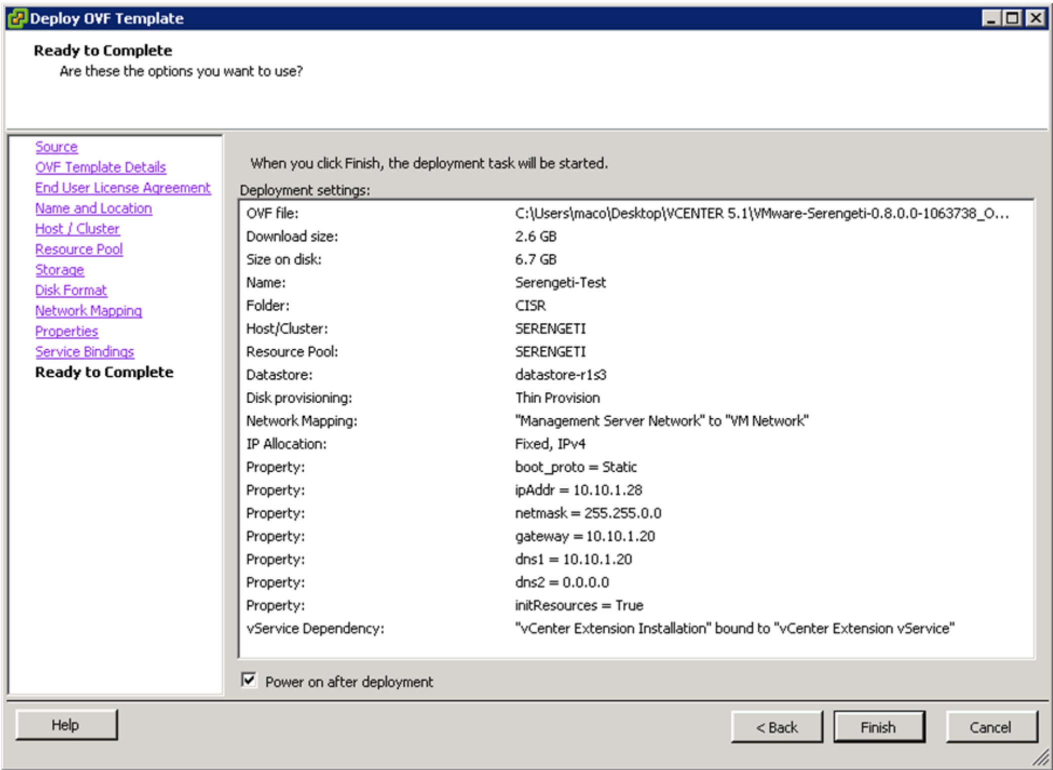


Figure 77. OVF Template Summary Page.

16. The Serengeti installation process will begin (see Figure 78). This process takes approximately 8–10 minutes to complete.

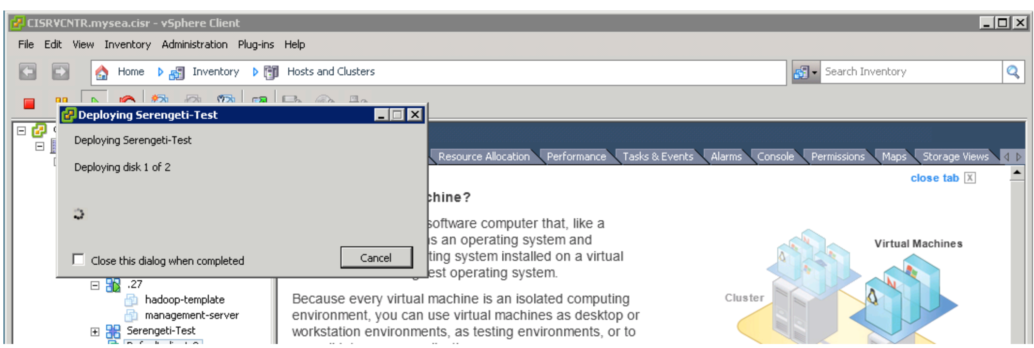


Figure 78. Serengeti Deployment Test Status Page.

17. Once installation is complete, the new Serengeti vApp will appear in vCenter. Included are two new virtual machines, the Serengeti Management Server, and the template VM used to clone the Hadoop clusters (see Figure 79).

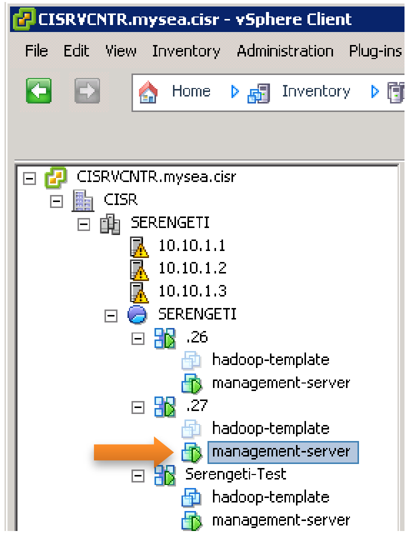


Figure 79. Serengeti vApp "Serenti-Test."

C. CREATE A HADOOP CLUSTER WITH DEFAULT SETTINGS

After a successful installation, it is recommended to deploy a Hadoop cluster using the application's default settings to ensure it is working properly.

Do the following:

1. Open the Serengeti Management Server Console by right clicking the **management-server** and selecting **Open Console** (see Figure 80).

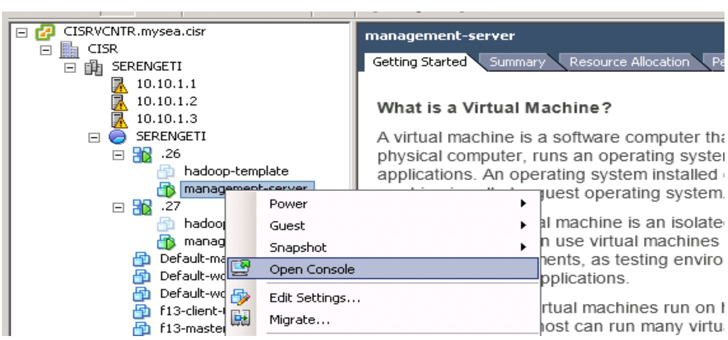


Figure 80. Open Console Menu Option.

2. Log in to the VM with the username `serengeti`. The default password will appear in the login banner, along with instructions on how to change the password after logging in.
3. From the console, Type `serengeti` at the command line to launch the Serengeti shell (see Figure 81).

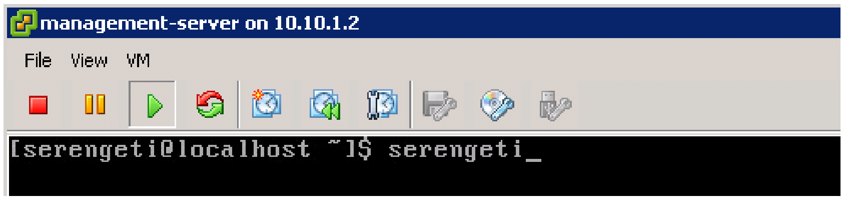


Figure 81. Serengeti Management Server Console.

4. Run the “connect” command to connect to the Serengeti server as follows (see Figure 82):

```
    connect --host <hostname or IP address>:8080 –username
<username> –password <password>
```

Note: The default username is `serengeti`, use default password.

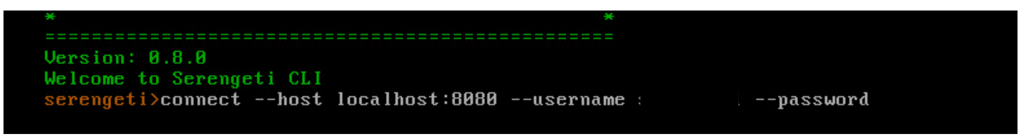


Figure 82. Serengeti “Connect” Command Syntax.

5. Run the “cluster create” command to deploy a Hadoop cluster (see Figure 83).

```
    cluster create –name <cluster name assigned by user>
```

Note: Only alphanumeric names (“a-z,” “A-Z”), numbers (“0–9”), and underscores (“_”) can be used in cluster name.

Figure 83. Serengeti “Cluster Create” Command Syntax.

6. This command will deploy a Hadoop cluster with 5 virtual machines: (1) Master Node, (3) Worker Nodes, and (1) Client Node (see Figure 84). Within 60 seconds of executing the command, the deployment process will start running and the virtual machines will begin to populate in vCenter (see Figure 85).

```
RUNNING 5%

node group: master,  instance number: 1
roles:[hadoop_namenode, hadoop_jobtracker]
  NAME          IP  STATUS     TASK
  -----------------------------------
  f13-master-0      Not Exist  Cloning

node group: worker,  instance number: 3
roles:[hadoop_datanode, hadoop_tasktracker]
  NAME          IP  STATUS     TASK
  -----------------------------------
  f13-worker-2      Not Exist  Cloning
  f13-worker-1      Not Exist  Cloning
  f13-worker-0      Not Exist  Cloning

node group: client,  instance number: 1
roles:[hadoop_client, pig, hive, hive_server]
  NAME          IP  STATUS     TASK
  -----------------------------------
  f13-client-0      Not Exist  Cloning
```

Figure 84. Cluster Creation Status.

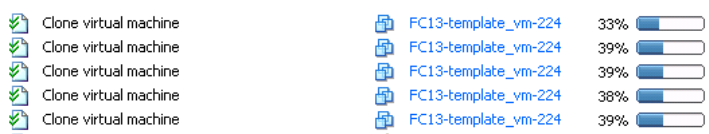


Figure 85. Virtual Machine Status Pane in vCenter.

7. Once the cluster is completed, Serengeti will indicate that all VMs are *Service Ready* and the cluster has been created (see Figure 86).

```
SUCCESS 100%

node group: master,  instance number: 1
roles:[hadoop_namenode, hadoop_jobtracker]
  NAME           IP           STATUS         TASK
  ----------------------------------------------
  f13-master-0  10.10.1.38  Service Ready

node group: worker,  instance number: 3
roles:[hadoop_datanode, hadoop_tasktracker]
  NAME           IP           STATUS         TASK
  ----------------------------------------------
  f13-worker-2  10.10.1.33  Service Ready
  f13-worker-1  10.10.1.34  Service Ready
  f13-worker-0  10.10.1.31  Service Ready

node group: client,  instance number: 1
roles:[hadoop_client, pig, hive, hive_server]
  NAME           IP           STATUS         TASK
  ----------------------------------------------
  f13-client-0  10.10.1.32  Service Ready

cluster f13 created
serengeti>_
```

Figure 86. Cluster Completion Status.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX J. CREATING A HADOOP TEMPLATE WITH FEDORA 13 AND FEDORA 18 OPERATING SYSTEMS

The Serengeti 0.8 vApp was released with CentOS 5.6 as the operating system for the Hadoop template VM. This Appendix provides instructions to create a Hadoop template with the Fedora 13 and 18 operating systems. Note that using these steps resulted in the successful provisioning of both Fedora 13 and Fedora 18 Hadoop clusters; however, only the Fedora 13 clusters were functional. The Fedora 18 clusters failed testing due to an error reported in Chapter VI. There may be additional configurations required in order to create a functional Hadoop cluster with Fedora 18.

A. CLONE HADOOP TEMPLATE FROM THE SERENGETI VAPP

1. Log on VADMIN1 server as an `Administrator` (see Appendix A, Table 8).
2. Double click on VMware vSphere Client icon on the Desktop and log into vCenter Server using `Administrator` credentials (see Appendix A, Table 8).
3. From vCenter, right click the Hadoop-template VM that was installed during the Serengeti OVA installation (see Appendix I), and select **Clone** (see Figure 87).

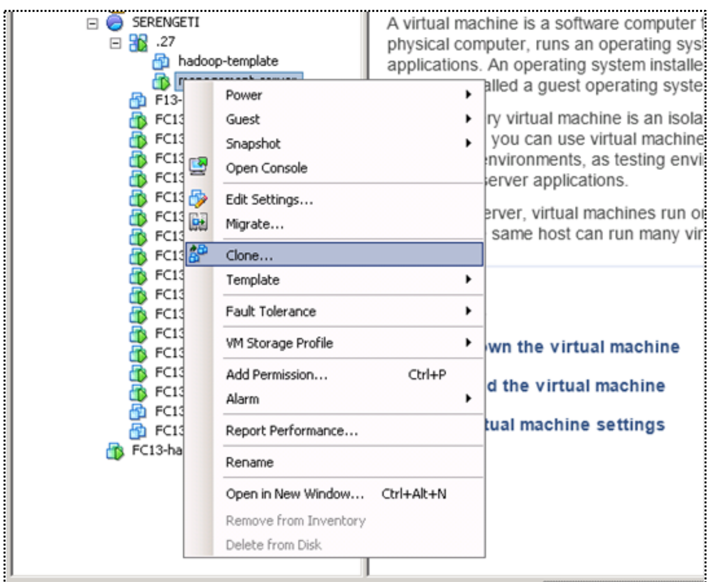


Figure 87. Clone Selection via Hadoop-Template VM.

Note: We cloned the Hadoop-template in order to use the default settings on the CentOS template as a guide. If this is not necessary you could simply proceed with section B of this guide and install Fedora 13 directly on this template.

4. Name the template, select the datacenter to store the VM in, and then click **Next** (see Figure 88).

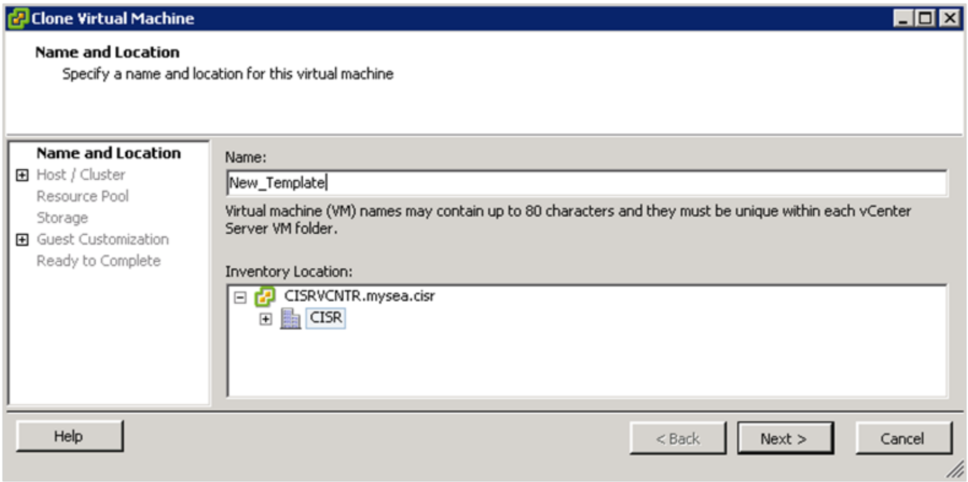


Figure 88. Clone VM Wizard – Name and Location Specification Page.

5. Select the cluster in which to store the VM, and then click **Next** (see Figure 89).

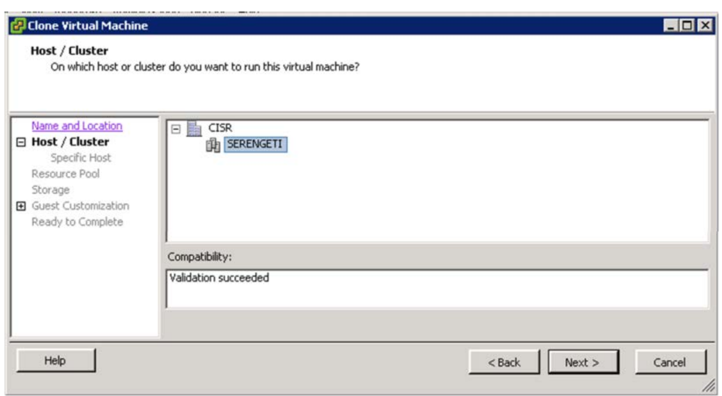


Figure 89. Clone VM Wizard – Host / Cluster Designation Page.

6. Select the resource pool in which to store the VM, and then click **Next** (see Figure 90).

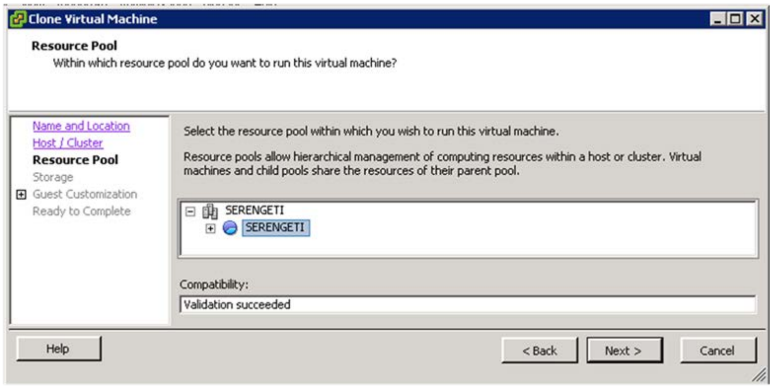


Figure 90. Clone VM Wizard – Resource Pool Designation Page.

7. Select the desired disk format and the datastore in which to store the VM, and then click **Next** (see Figure 91).

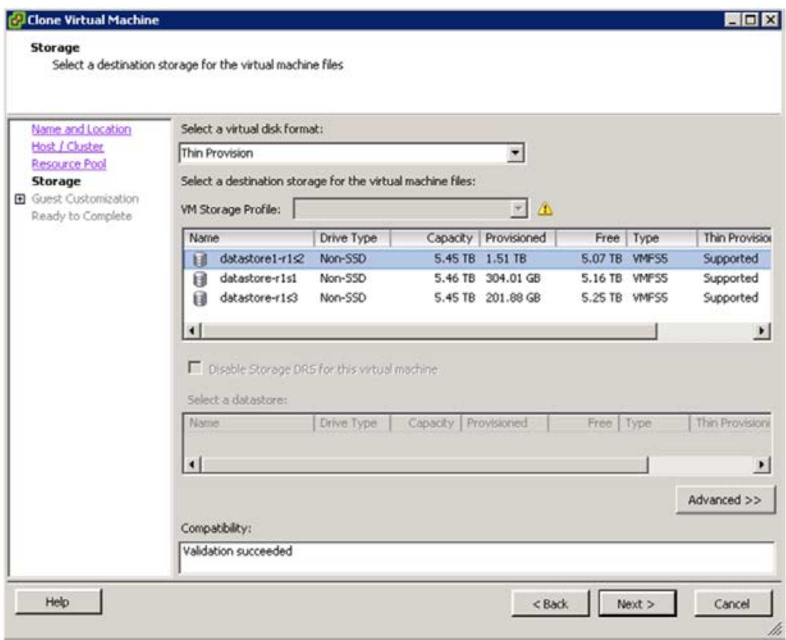


Figure 91. Clone VM Wizard – Storage Designation Page.

8. Ensure *Power on this virtual machine after creation* is NOT checked (see Figure 92), and then click **Next,**

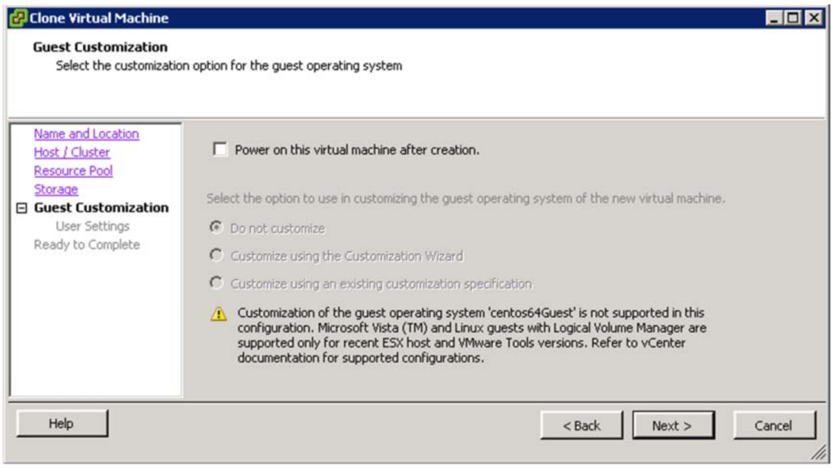


Figure 92. Clone VM Wizard – Guest Customization Page.

9. Confirm settings (see Figure 93) and click **Finish.**

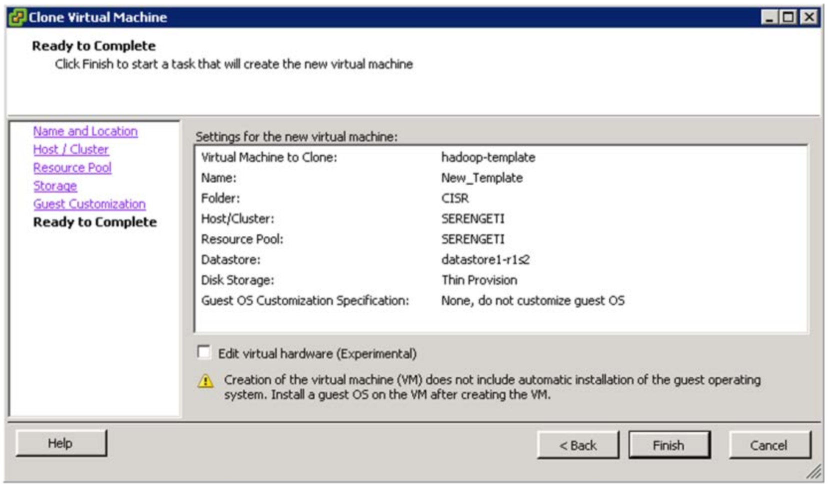


Figure 93. Clone VM Wizard – New VM Summary Page.

Note: Cloning may take several minutes. You can track the status of the cloning process in the vCenter Recent Tasks pane. Cloning must be completed in order to proceed with step B of this guide.

B. INSTALL FEDORA ON CLONED TEMPLATE

Do the following:

1. From VADMIN1, download the Fedora ISO.
   1.1. Fedora 13 and Fedora 18 are available from the Fedora Project archive[10].

2. Upload the Fedora ISO to a datastore using the steps listed in Appendix L section B.
3. Map the Fedora ISO using the steps listed in Appendix L section C steps 1–5. On step 6 confirm that both Connected and Connect at power on are selected, but DO NOT click OK.
4. Select the **Options** tab, select **Boot Options** from the left pane, then check the **The next time the virtual machine boots, force entry into the BIOS setup screen** option (see Figure 94). Click **OK** to continue.

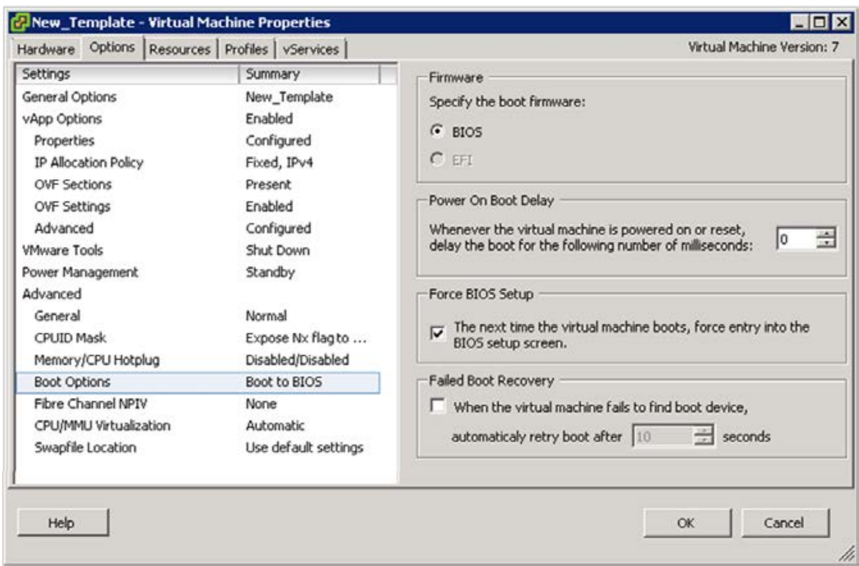


Figure 94. VM Properties – Boot Options Page.

5. Right click on the VM, and select **Open Console** (see Figure 95).

[10] Available online at: http://archive.fedoraproject.org/pub/archive/fedora/linux/releases/13/Fedora/x86_64/iso.

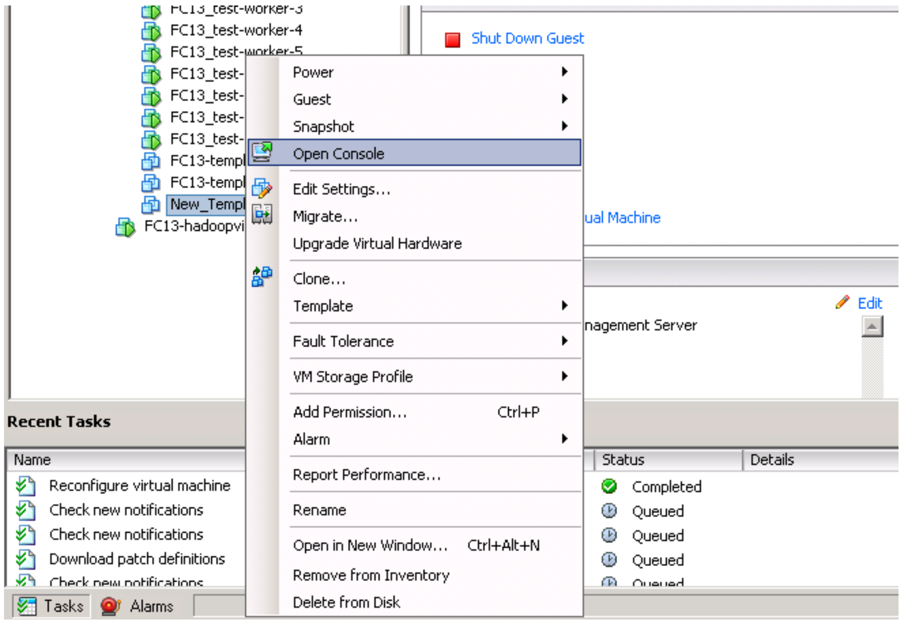


Figure 95. Open Console Menu Option.

6. Click the green Power On button to start the VM.
7. Using the controls listed on the screen, scroll to the Boot tab and move **CD-ROM Drive** to the top of the boot order (see Figure 96).

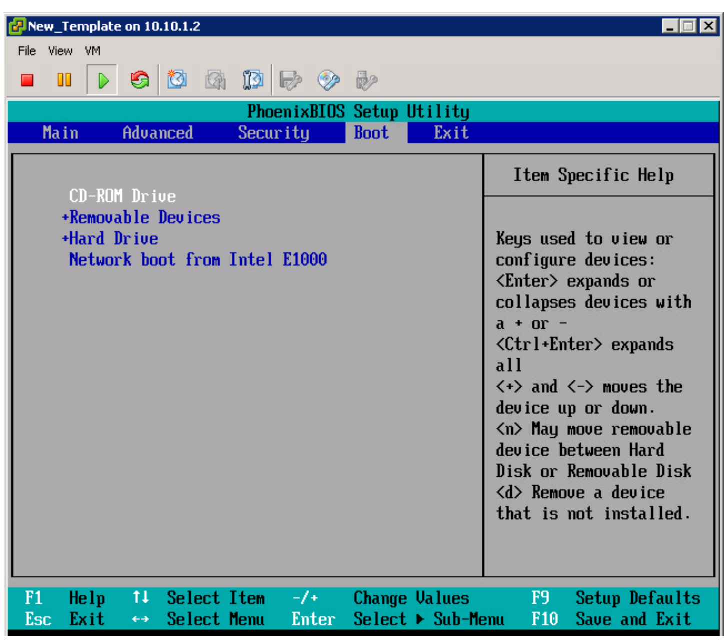


Figure 96. VM BIOS Setup Utility – Boot Options Tab.

8. Scroll to the Exit tab and select **Exit Saving Changes**, and then select **Yes** to confirm (see Figure 97).

Figure 97. VM BIOS Setup Utility – Setup Confirmation Window.

9. To install Fedora 13, do the following:

9.1. Select Install a new system or upgrade an existing system.
9.2. Select **Skip** to bypass the media test.
9.3. Click **Next.**
9.4. Click **Next** to select English as the default language.
9.5. Click **Next** to select English as the system keyboard.
9.6. Click **Next** to select Basic Storage Devices.
9.7. Click **Next** to accept the default Hostname.
9.8. Select the appropriate time zone, and then click **Next.**
9.9. Enter the desired password for the Root account and then click **Next.**
9.10. Select **Use All Space**, and then click **Next.**
9.11. Click Write changes to disk.
9.12. Select **Minimal** installation, use the **Installation Repo**, and then click **Next.**
9.13. Click Reboot. Fedora 13 will boot to the login screen.

10. To install Fedora 18, do the following:

10.1. Select Install Fedora.
10.2. Choose English (United States) as the system language.
10.3. Under LOCALIZATION click **DATE & TIME**.
10.3.1. Select the correct time zone and then click **Done**.
10.4. Under SOFTWARE click **SOFTWARE SELECTION**.
10.4.1. Under Choose your environment, select **Minimal Install**.
10.4.2. Under Choose your add-ons, select **Standard**.
10.4.3. Click **Done**.
10.5. Under STORAGE click **INSTALLATION DESTINATION**.

10.5.1. Ensure the VMware Virtual disk is highlighted in blue (click on it if it is not), and then click **Continue**.
10.5.2. Click **Reclaim space**.
10.5.3. Mark each of the filesystems for deletion by clicking on them and then clicking **Delete**.
10.5.4. Click **Reclaim space**.
10.5.5. Click **Begin Installation**.
10.5.6. A warning will appear that indicates the Root password has not been set, click the warning, enter the password and click Done.
10.5.7. Click Reboot.
10.5.8. After the first reboot, Fedora 18 will boot to the Main Menu. Select **Troubleshooting**.
10.5.9. Select **Boot from local drive**. Fedora 18 will now boot to the login screen.

C. CONFIGURE THE TEMPLATE

Note: Some of the steps are performed differently depending on which operating system is installed (Fedora 13 or Fedora 18). When there is a difference it will be indicated at the beginning of the step.

1. Log in to the VM as root via VMware console.
2. **(Fedora 13 only)** Configure network communications by editing the `ifcfg-eth0` file.
   2.1. Enter command: `vi /etc/sysconfig/network-scripts/ifcfg-eth0`
   2.2. Remove the `HWADDRESS`, change `ONBOOT` to `=yes`, and add `BOOTPROTO=dhcp`: (Final changes are reflected in Figure 98).

Figure 98. Ethernet Controller 0 Interface File.

   2.3. Save and exit by typing `:wq`.
3. **(Fedora 13 only)** Restart networking by entering the command `service network restart.`
4. Confirm network connectivity by using the `ifconfig` command; `eth0` should be listed with an IP address (see Figure 99). If the template does not pick up an IP

address, ensure network connectivity was configured in accordance with Appendix E and H.

Note: PuTTY (shown in Figure 99) PuTTY is an SSH and telnet client which can be used to establish a remote console session to the VM after networking is configured. Using PuTTY can save time by allowing the user to copy and paste necessary commands from an electronic (soft-copy) reference vice typing them manually.

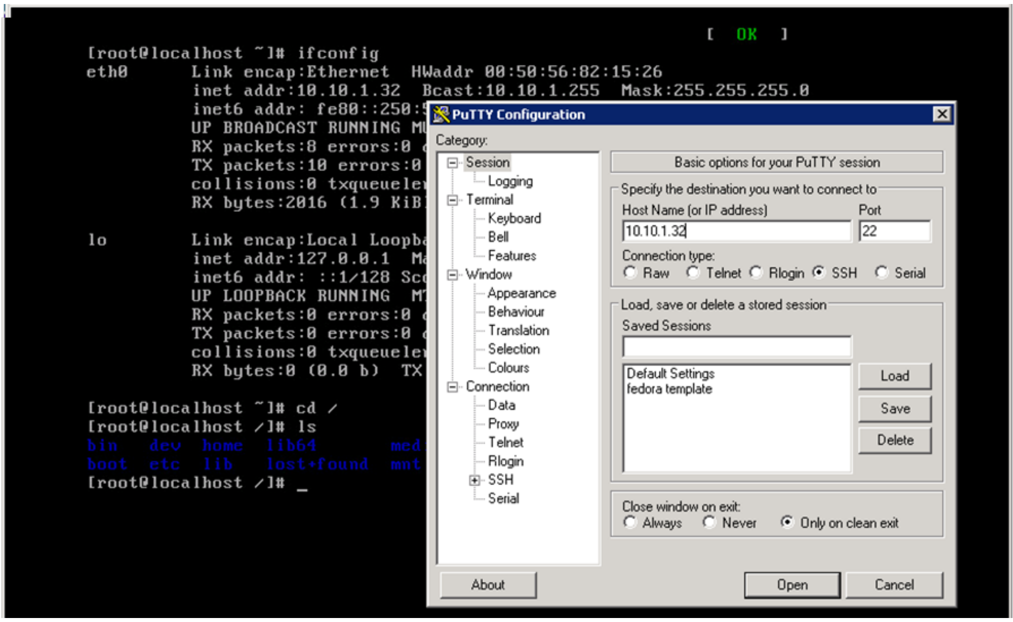


Figure 99. PuTTy Configuration Page.

5. Download the Serengeti Installation Guide[11]
6. Scroll down to the **Instruction for Creating Serengeti Node Template** section. There are four sections of this installation guide that require further details than what is provided or need to be modified for Fedora 13:
    - Add serengeti user and make it as sudoer without password
    - Install Sun JRE 1.6 or JDK 1.6
    - Add agent scripts

[11] Available online at: https://github.com/vmware-serengeti/doc/blob/master/installation_guide_from_source_code_M2.md.

- Override ifcfg-eth0 to avoid NIC brought online by the network service **(Fedora 13 only)**

Additionally, this Appendix includes three additional requirements not listed in the installation guide:

- Install postgresql
- Install VMware Tools
- Delete the /etc/udev/rules.d/70-persistent-net.rules **(Fedora 13 only)**

These tasks are described in section D-F of this Appendix.

6.1. **(Fedora 18 only)** From the console of the template, enter the following command: `yum remove audit.x86_64`

6.2. Perform the "yum install following packages" step as listed.

6.3. Perform the "reduce grub boot waiting time" step **(OS dependent)**:

6.3.1. **(Fedora 13)** Perform step as listed.

6.3.2. **(Fedora 18)** Use the following command:

```
sed -i 's|^timeout=.*$|timeout=0|'/boot/grub2/grub.cfg
```

6.4. Perform the "add write permission to /tmp directory" step as listed.

6.5. Perform the "install ruby 1.9.2" step as listed.

6.6. Perform the "install chef and its dependencies" step as listed.

6.7. Perform the "add serengeti user and make it as sudoer without password" step as follows:

6.7.1. Perform the steps as listed, and repeat to create an additional user account. Repeat them in order to create and additional user account (See Appendix A, Table 8), which you will use to log in to the Hadoop VMs after their creation. The `serengeti` and `root` account passwords will change during cluster creation and you will not be able to log in to the VMs with those accounts.

6.8. Prepare to perform the "install Sun JRE 1.6 or JDK 1.6" step as follows:

> **Note**: The installation guide states to upload the JRE installation package to the /root directory, but does not specify how to perform this.

6.8.1. From VADMIN1, download `jre-6u31-linux-x64-rpm.bin`[12]

6.8.2. Perform all steps in Appendix L to create an ISO image of this file, upload it to the datastore, and map it to the VM's CD-ROM drive.

6.8.3. From the console of the template, perform the following steps:

[12] Available online at: http://www.oracle.com/technetwork/java/javasebusiness/downloads/java-archive-downloads-javase6-419409.html#jre-6u31.

```
cd /
cd /tmp
mkdir cdrom
mount /dev/cdrom /tmp/cdrom
cd cdrom
cp * /root
```

6.9. Perform the "install SUN JRE 1.6 or JDK 1.6" step as listed.

6.10. Add Agent Scripts

Note: The installation guide states to copy distribute/agent/* under serengeti-ws github repo to the /opt/vmware/sbin, but does not specify how to perform this.

6.10.1. From VADMIN1, download the agent scripts.

6.10.2. The agent scripts are available from the Serengeti source code page[13].

6.10.3. Download the following files:

- get_json_value.py
- machine_id_guest_var
- mount_swap_disk.sh
- setup-ip.py

6.10.4. Create an ISO image of this file, upload it to the datastore, and map it to the VM's CD-ROM drive in accordance with Appendix L.

6.10.5. From the console of the template, perform the following steps:

```
mkdir –p /opt/vmware/sbin
cd /
cd /tmp
umount /tmp/cdrom
```
(*if cdrom was previously mounted in /tmp from JRE install*)

```
mount /dev/cdrom /tmp/cdrom
```
(*if you receive an error at this step, disconnect and reconnect the cdrom drive in the VM's edit settings menu*)

```
cd cdrom
cp * /opt/vmware/sbin
echo “python /opt/vmware/sbin/setup-ip.py” >>
/etc/rc.local
echo “bash /opt/vmware/sbin/mount_swap_disk.sh” >>
/etc/rc.local
```

6.10.6. <u>DO NOT</u> perform the "override ifcfg-eth0 to avoid NIC brought by network service" step.

[13] Available online at: https://github.com/vmware-serengeti/serengeti-ws.

Note: Changing the `ifcfg-eth0` on the Fedora template will prevent the Hadoop VMs from acquiring an IP address from the DHCP server. On Fedora 13, leave `ifcgf-eth0` as configured in Section C, steps 2.2–2.3 of this Appendix. There is no modification of `ifcfg-eth0` required on Fedora 18.

6.10.7. (Fedora 13 only) Perform "**stop firewall**" steps as listed.
6.10.8. Perform "**disable selinux in /etc/selinux/config**" step as listed.

D. INSTALL POSTGRESQL

1. From the template console, perform the following steps:

```
yum install postgresql
yum install postgresql-server
yum install postgresql-jdbc
service postgresql initdb
chkconfig postgresql on
```

E. INSTALL VMWARE TOOLS

Note: VMware Tools must be installed on the template in order for the cloning process to work properly.

1. Click **VM** and select **Guest** then select **Install/Upgrade VMware Tools** (see Figure 100).

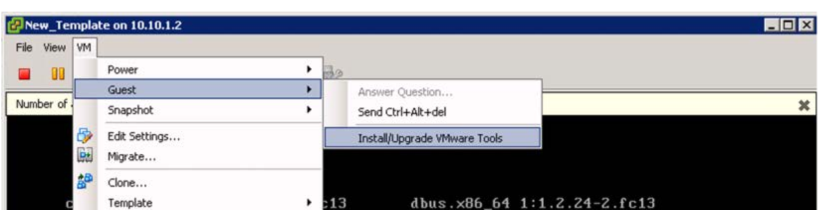


Figure 100. VMware Tools Installation Menu Selection.

2. Click **OK** on the Install VMware Tools banner.
3. From the template console, perform the following steps:

`cd /`
`cd /tmp`
`umount /tmp/cdrom` *(if cdrom was previously mounted in /tmp from agent scripts install)*
`mount /dev/cdrom /tmp/cdrom`

```
cd cdrom
cp VM* /tmp
cd ..
tar xzvf VM (press tab to autocomplete)
cd vmware-tools-distrib
./vmware-install.pl
```

4. Press enter to accept each default installation setting.
5. To confirm that VMware Tools is installed, check the Summary Tab on the Template VM in vCenter (see Figure 101).

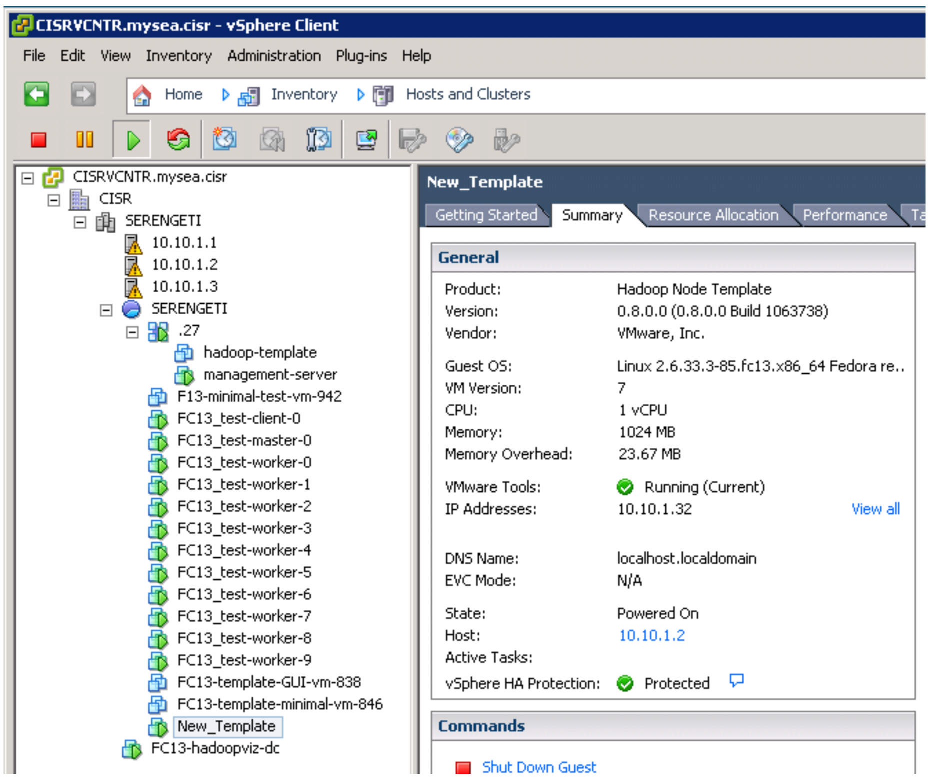


Figure 101. VM Template Summary Tab.

F. DELETE 70-PERSISTENT-NET.RULES AND SHUTTING DOWN THE VM

Note: Deleting `70-persistent-net.rules` prevents issues with the cloned VMs recognizing `eth1` as the primary adapter vice `eth0`. Not performing this step will cause the cluster creation to fail. It is only performed on the Fedora 13 template.

1. **(Fedora 13 only)** From the template console, perform the following steps:

```
cd /
cd /etc/udev/rules.d
rm –rf 70-persistent-net.rules
```

2. Type `shutdown –h 0` to shutdown the VM.
3. Click **File** and select **Exit** to close the VM console.

G. CONFIGURE SERENGETI TO USE THE NEW TEMPLATE

The Serengeti Management Server uses the serengeti.properties file to identify the VM to use as the Hadoop template. In order to modify this file, you must first identify the virtual machine ID of the template you created.

1. To find the VM number, enter the vCenter IP address `(10.10.1.10)` in a web browser. Click **Browse objects managed by this host** in the lower right-hand corner of the page (see Figure 102).

Figure 102. vSphere Web Homepage.

2. Enter the vCenter user name and password when prompted (see Appendix A, Table 8).
3. The ServiceInstance page will open, in the Methods table under the NAME column, Click **RetrieveServiceContent.**
4. The RetrieveServiceContent page will open, click **Invoke Method.**

5. In the Method Invocation Result: **ServiceContent table**, locate the row that contains the values shown in Table 10, and then click the link in the Value column:

Note: text in VALUE column may vary based on vCenter configuration.

NAME	TYPE	VALUE
rootfolder	ManagedObjectReference:Folder	group-d1 (Datacenters)

Table 10. Service Content Table

6. The group-# page will open. In the Properties table, locate the row that contains the values shown in Table 11, and then click the link in the Value column.

NAME	TYPE	VALUE
childEntity	ManagedObjectReference:ManagedEntity	datacenter-2 (CISR)

Table 11. Group Number Properties Table

7. The datacenter page will open. In the Properties table, locate the row that contains the following values shown in Table 12, and in the VALUE column click the link for the `datastore-r1s1` (designated datastore from Section A, step 7 of this Appendix).

NAME	TYPE	VALUE
datastore	ManagedObjectReference:Datastore	datastore-### (datastore1-r1s1) datastore-### (datastore1-r1s2) datastore-### (datastore1-r1s3)

Table 12. Data Center Properties Table

8. The datastore page will open, scroll down to the last row of the properties table. You will see the list of VMs on the datastore with their `vmid`. Find the name of the template you created and take note of the associated `vmid` (see Figure 103).

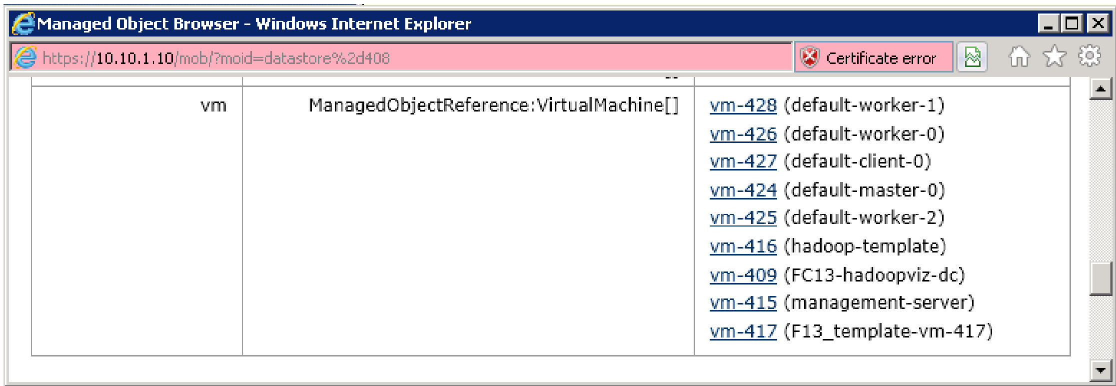


Figure 103. Managed Object Browser.

H. UPDATE THE SERENGETI.PROPERTIES FILE

1. Right click on the Serengeti Management Server VM and select **Open Console.**
2. Log into the Serengeti Management server (see Appendix I, section C, step 2).
3. Open the `serengeti.properties` file in VI editor:

```
sudo vi /opt/serengeti/conf/serenegeti.properties
```

4. Edit the file to change the template_id to the number of the newly created template (see Figure 104).

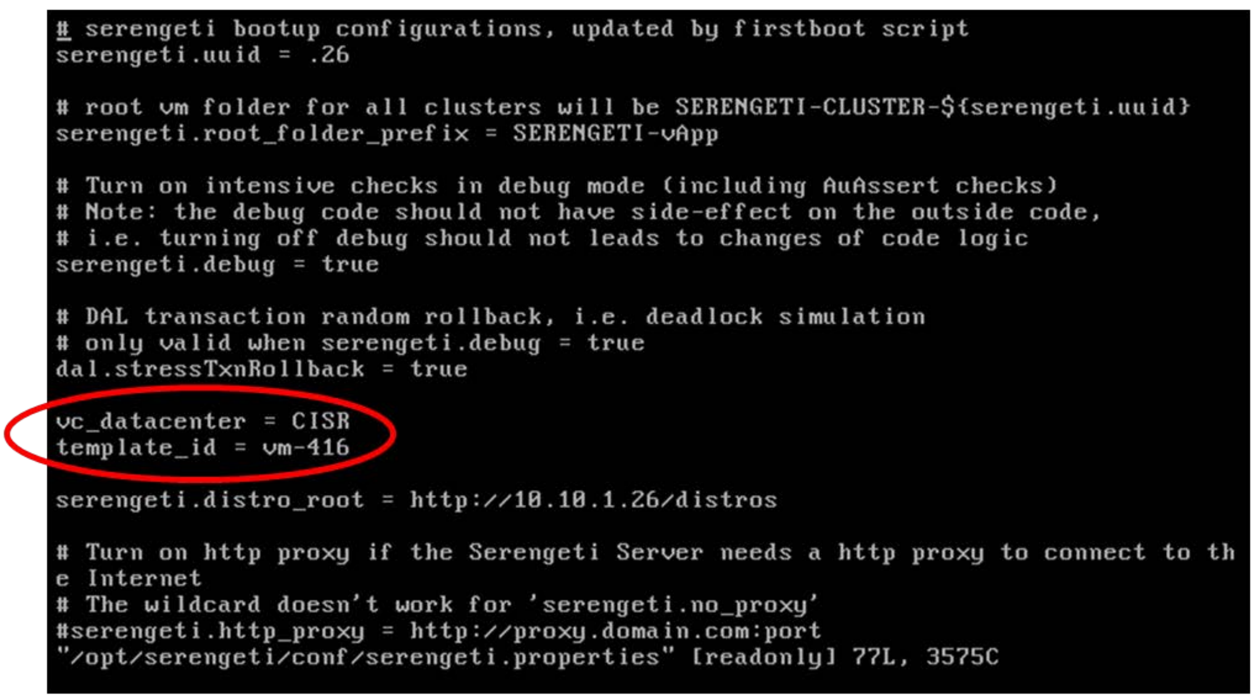


Figure 104. Serengeti.Properties File.

5. Save and exit serengeti.properties file.
6. Restart Serengeti services:

```
serengeti-stop-services.sh
serengeti-start-services.sh
```

I. CREATE A FEDORA 13 HADOOP CLUSTER

1. Follow the steps in Appendix J, Section I to create a cluster in Serengeti.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX K. CREATING CHEF'S COOKBOOK

A. PRELUDE

By default, the Serengeti virtual appliance is designed to utilize CentOS 5.6 as a template for provisioning Hadoop cluster nodes. In order to enable the deployment of Fedora 13-based nodes, the Hadoop virtual machine template has to be configured in accordance with Appendix J and Chef's configuration requires adjustment as well. Chef is an application that relies on reusable definitions known as *cookbooks* and *recipes* written using the Ruby programming language. Cookbooks and recipes automate common infrastructure tasks during the deployment of Hadoop cluster via Serengeti. Their definitions describe what your infrastructure consists of and how each part of your infrastructure should be deployed, configured and managed. Chef applies those definitions to servers (nodes) to produce an automated infrastructure.

B. MODIFICATION

The Serengeti development team has created the cookbooks to support CentOS 5.6, but they are incompatible with Fedora 13. Therefore, we have modified Serengeti-pantry [22] to resolve this problem by performing the following steps:

1. Logon Serengeti management server via ssh (see Figure 105), or VMware console as user **Serengeti** (see Appendix A, Table 8).

Note: We have configured the address translation on DC01 to redirect external & inbound ssh traffic to the internal Serengeti management server. Therefore, one can use DC01's external IP address to ssh to Serengeti management server.

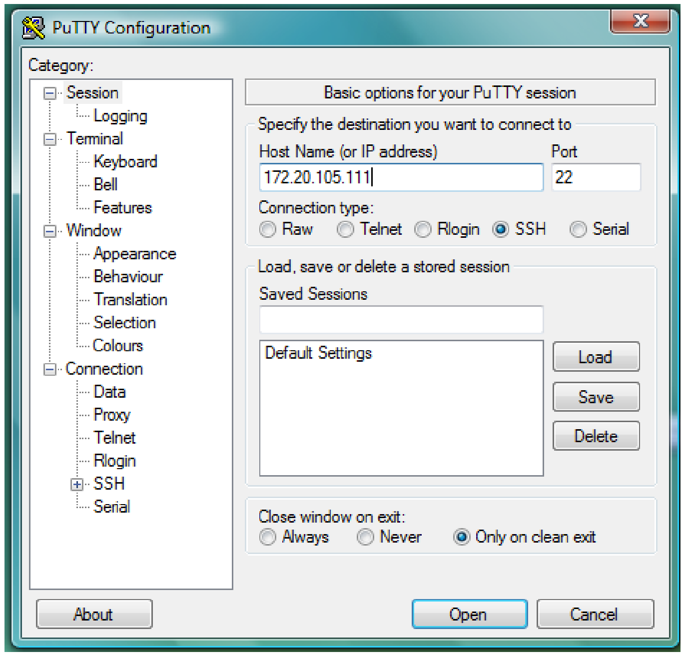


Figure 105. PuTTy Configuration Settings.

2. Modify **hadoop_common** cookbook.

2.1. Navigate to **hadoop_common/recipes** directory by issuing the following command:

```
$ cd  >
/opt/serengeti/cookbooks/cookbooks/hadoop_common/recipes
```

Note: Use the `cat -n add_repo.rb` command if necessary to number the lines in the "add_repo" recipe.

2.2. Issue the following command:

```
$ sudo vi add_repo.rb
```

2.3. Replace lines 21 thru 23 with the following lines:

```
21  case node[:platform]
22  when 'centos', 'fedora', 'redhat'
23  prefix = node[:platform] == 'centos' ? 'Fedora' : 'rhel'
```

2.4. Verify the changes, then save and quit the recipe by typing "`:wq`."

3. Modify **hive** cookbook.

3.1. Navigate to **hive/recipes** directory

```
$ cd /opt/serengeti/cookbooks/cookbooks/hive/recipes
```

Note: Use the `cat –n postgresql_metastore.rb` command if necessary to number the lines in the “postgresql_metastore” recipe.

3.2. Issue the following command:

```
$ sudo vi postgresql_metastore.rb
```

3.3. Replace line 23 with the following line

```
23   cp /usr/share/java/postgresql-jdbc-8.4.701.jar #

{node[:hive][:home_dir]}/lib/
```

3.4. Verify the changes, then save and quit the recipe by typing “`:wq`.”

4. Modify **hadoop_cluster** cookbook.

Note: This modification is required only to support Hadoop visualization software and it is not affecting the deployment of a Hadoop cluster.

4.1. Navigate to hadoop_cluster/templates/default directory

```
$ cd /opt/serengeti/cookbooks/cookbooks/hadoop_cluster/templates

/default
```

Note: Use the `cat –n log4j.properties.erb` command if necessary to number the lines in the “log4j.properties” template.

4.2. Issue the following command:

```
$ sudo vi log4j.properties.erb
```

4.3. Replace line 4 with the following line

```
4  hadoop.root.logger=<%= conf[‘hadoop.root.logger’] ||

‘INFO,RFA,SYSLOGM’ %>
```

4.4. Scroll down to the last line and append the file with the following set of lines:

```
#
#HadoopViz Appender
#
log4j.appender.SYSLOGM=org.apache.log4j.net.SyslogAppender
log4j.appender.SYSLOGM.facility=local1
log4j.appender.SYSLOGM.layout=org.apache.log4j.PatternLayou
t
log4j.appender.SYSLOGM.layout.ConversionPattern=%p %c{2}: %
m%n
log4j.appender.SYSLOGM.SyslogHost=10.10.1.28:5679
```

```
log4j.appender.SYSLOGM.theshold=INFO
log4j.appender.SYSLOGM.FacilityPrinting=true
log4j.appender.SYSLOGM.Header=true
log4j.logger.org.apache.hadoop.hdfs.server.datanode.DataNod
e=INFO,SYSLOGM
```

4.5. Verify the changes, then save and quit the recipe by typing "`:wq`."

5. Execute the following command to apply all changes:

```
$ knife cookbook upload -a
Uploading cluster_service_discovery                 [0.1.0]
Uploading hadoop_cluster                            [1.2.0]
Uploading hadoop_common                             [0.1.0]
Uploading hbase                                     [0.1.0]
Uploading hive                                      [3.0.4]
Uploading install_from                              [3.0.4]
Uploading java                                      [2.0.0]
Uploading mapr                                      [0.1.0]
Uploading mysql                                     [1.2.4]
Uploading pig                                       [3.0.4]
Uploading postgresql                                [0.99.4]
Uploading tempfs                                    [0.1.0]
Uploading zookeeper                                 [0.1.0]
upload complete
```

5.1. Restart the Serengeti management server using the following commands:

```
$ cd /
$ cd /opt/serengeti/sbin
$ sudo ./serengeti-stop-services.sh
$ sudo ./serengeti-start-services.sh
```

APPENDIX L. ISO MANAGEMENT

Software that is manually installed on a virtual machine in VMware, such as operating systems, must be in optical disk image (ISO) format[14], which is indicated by a .iso file extension. Some software, such as Fedora operating systems, can be downloaded as an ISO, and directly uploaded to a datastore in vCenter. Other software, such as Microsoft operating systems, may need to be converted to an ISO before being uploaded. This Appendix covers converting software files to ISO format and uploading them to a datastore.

A. CONVERTING TO ISO FORMAT

1. Download or copy the target software package or file(s) to VADMIN1.
2. From VADMIN1, launch MagicISO (**Start>All Programs>MagicISO**).
3. Right-click in the right pane, and select **Add Files** (see Figure 106).

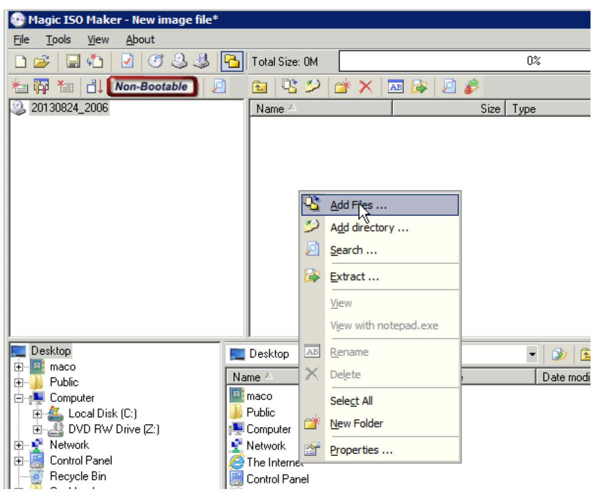


Figure 106. Magic ISO New Image Pane.

4. From Windows Explorer, navigate to the desired file(s), and then select **Open**.
5. Click the **Save** button in the toolbar (see Figure 107).

[14] One exception to this: VMware Open Virtual Appliance (OVA) files, such as Serengeti, do not need to be converted to ISO files before uploading.

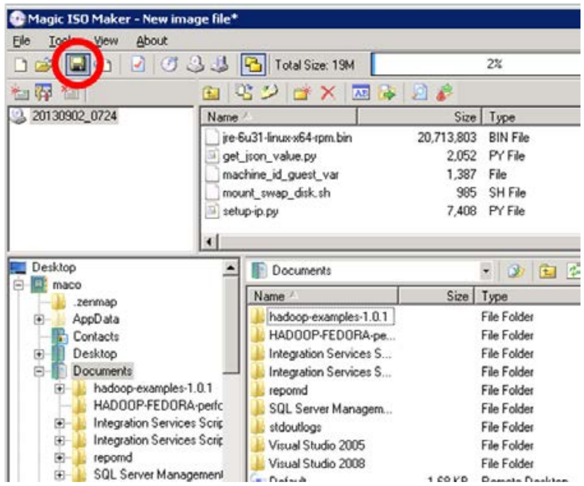


Figure 107. Save File(s) as ISO.

6. In the Save As… window, navigate to the desired directory and enter the desired filename. Ensure **Standard ISO Image (*.ISO)** is selected as the Format, and then click **Save**.

B. UPLOADING ISO TO A DATASTORE

Datastores are the storage locations used by ESXi hosts. In this test bed, the datastores consist of the local storage in each of the ESXi hosts. These datastores cannot be shared by multiple ESXi hosts, so when uploading software for installation on a VM, the software must be uploaded to the datastore associated with the VM.

1. In the left-hand pane in vCenter, click on the host to which the ISO will be uploaded.
2. Click the **Configuration** tab.
3. Under Hardware, click **Storage.**
4. Right-click the datastore, and select **Browse Datastore** (see Figure 108).

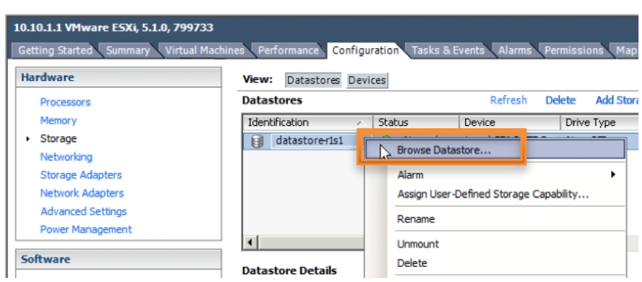


Figure 108. Browse Datastore Menu Option.

5. Click the “upload to datastore” button and select **Upload File** (see Figure 109).

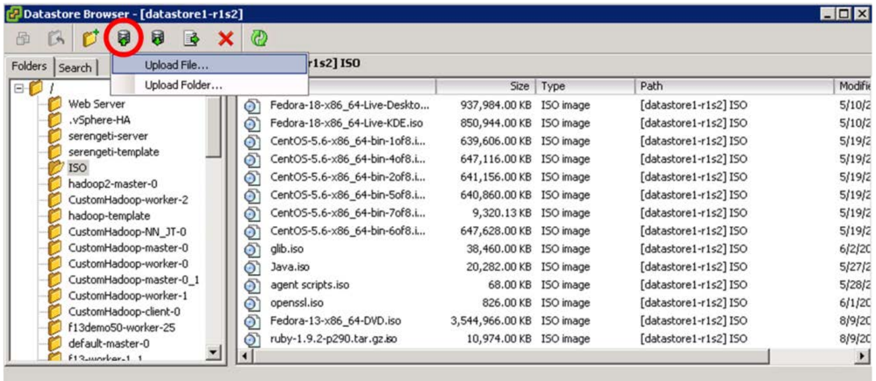


Figure 109. Upload to Datastore Selection Menu.

6. Navigate to the location of the ISO, and then click **Open** (see Figure 110).

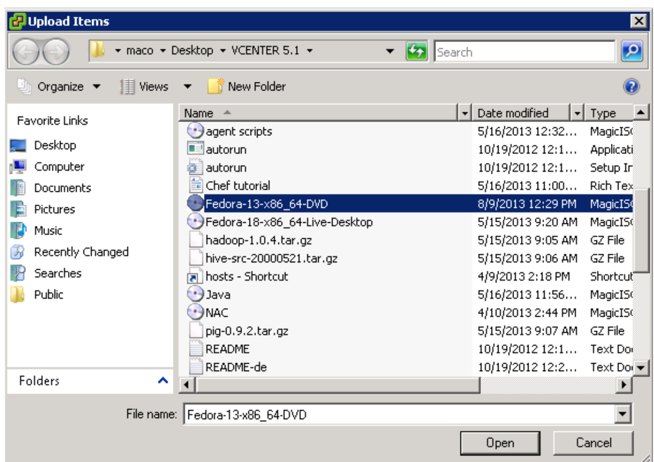


Figure 110. Upload Items Browser.

7. An Upload/Download Operation Warning will appear, click **Yes** to proceed. The upload process may take a few minutes, depending on the size of the ISO.

C. MAP ISO TO A VIRTUAL MACHINE’S CD-ROM DRIVE

1. In the left-hand pane of vCenter, right-click the VM on which to install the software and select **Open Console** (see Figure 111).

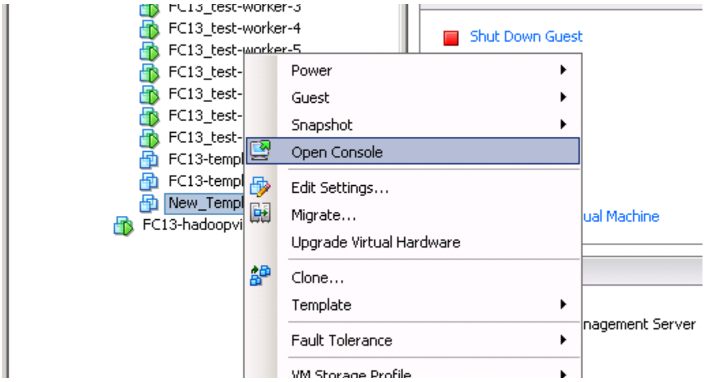


Figure 111. Opening the VM Console.

2. From the VM console, click **VM**, and select **Edit Settings** (see Figure 112).

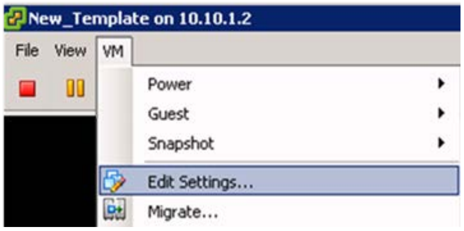


Figure 112. Edit Setting Menu Option in VM Console.

3. If **CD/DVD drive 1** is <u>NOT</u> included in the hardware list, perform the following steps; if it is already listed, skip to step 4:
   3.1. In the Hardware tab, Click **Add** (see Figure 113).

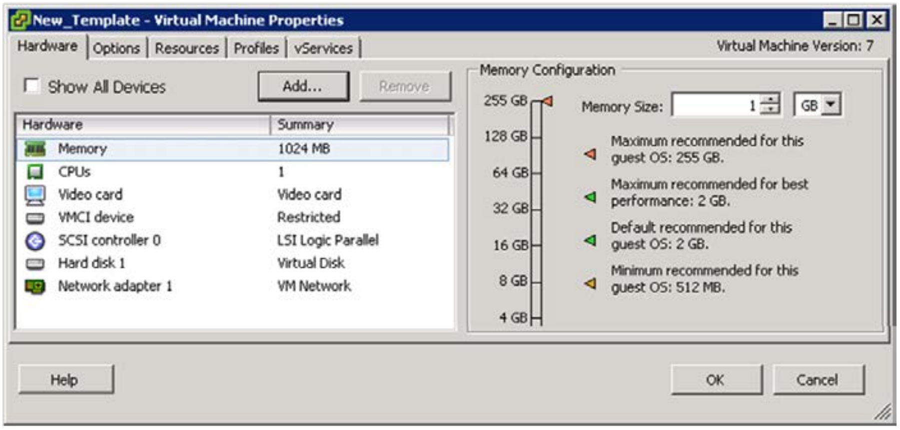


Figure 113. New Template VM Properties Page.

3.2. Select **CD/DVD Drive**, and then click **Next** (see Figure 114).

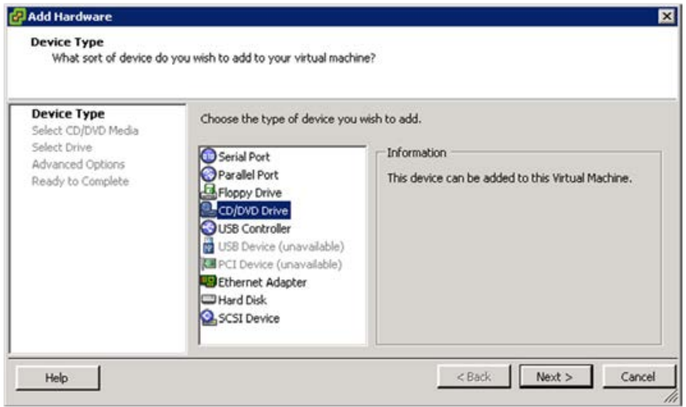


Figure 114. Add Hardware Device Selection Page.

3.3. Select **Use ISO image**, and then click **Next** (see Figure 115).

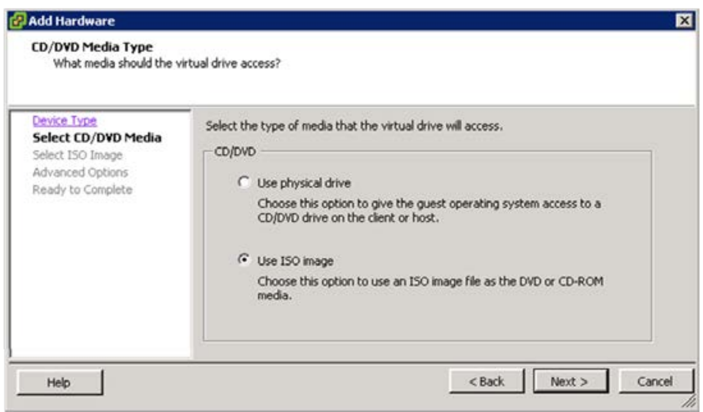


Figure 115. Add Hardware CD/DVD Selection Page.

3.4. Click **Browse.**

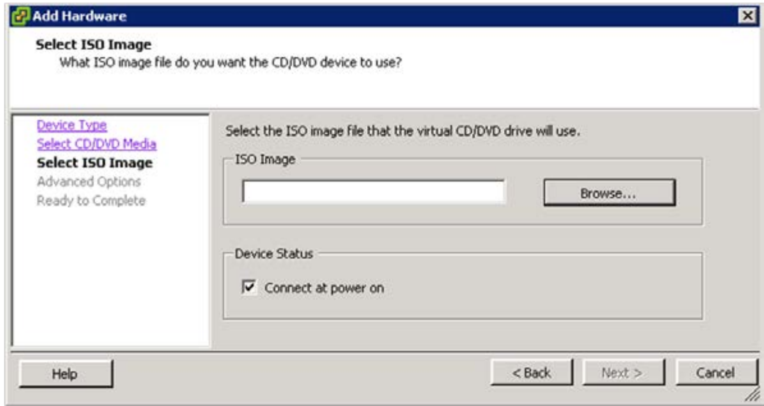


Figure 116. Add Hardware ISO Selection Page.

3.5. Navigate to the desired ISO file (see Figure 117), select it and click **OK.**

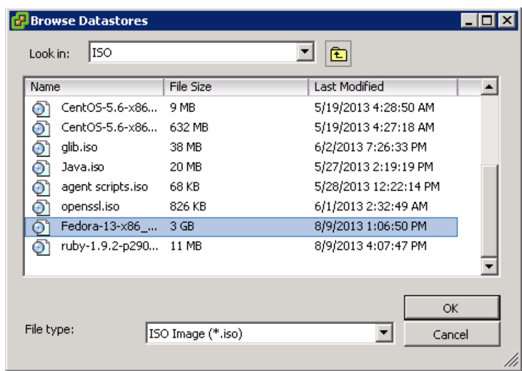


Figure 117. Datastore Browser Page.

3.6. Ensure **Connect at power on** is selected (see Figure 118), and then click **Next.**

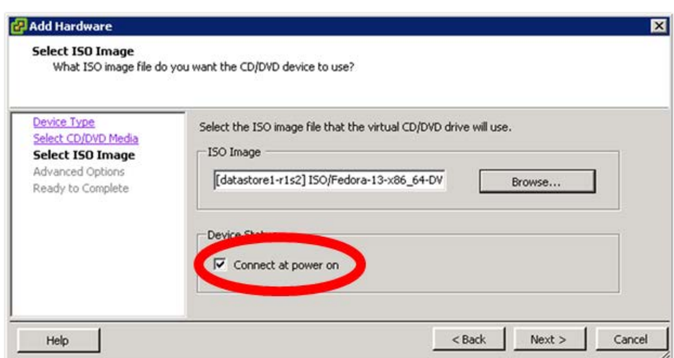


Figure 118. Add Hardware ISO Image Selection Page.

3.7. Click **Next** to accept the default Virtual Device node (see Figure 119).

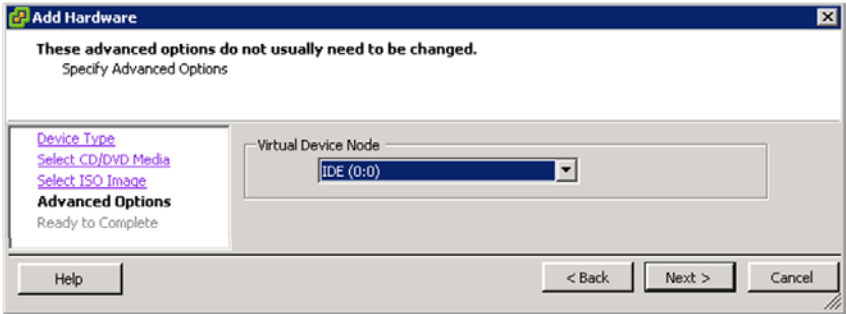


Figure 119. Add Hardware Advanced Options Page.

3.8. Verify the selected options, then Click **Finish** (see Figure 120)**.**

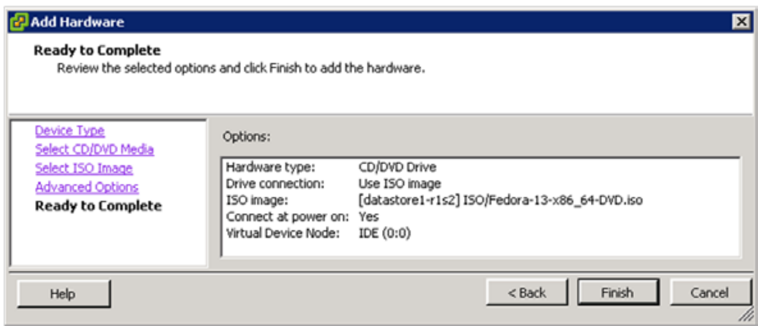


Figure 120. Add Hardware Review Selected Options Page.

4. Click on **CD/DVD drive 1** in the hardware list; ensure **Datastore ISO File** is selected, and then click **Browse** (see Figure 121).

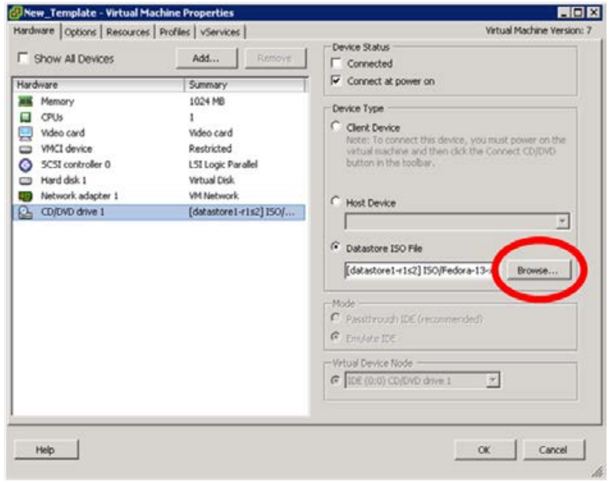


Figure 121. VM Hardware Properties Page.

5. Navigate to the desired ISO file, select it and click **OK (**see Figure 122).

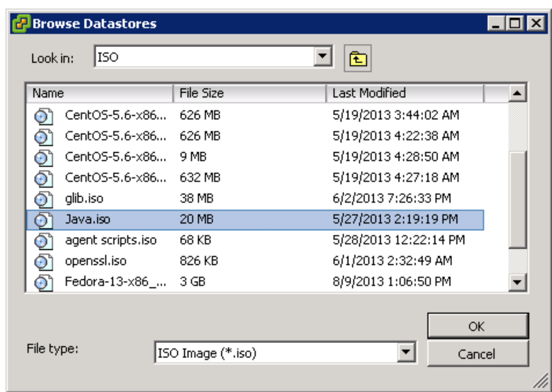


Figure 122. Datastore Browser Page.

6. Ensure **Connected** and **Connect at power on** are selected (see Figure 123), and then click **OK.**

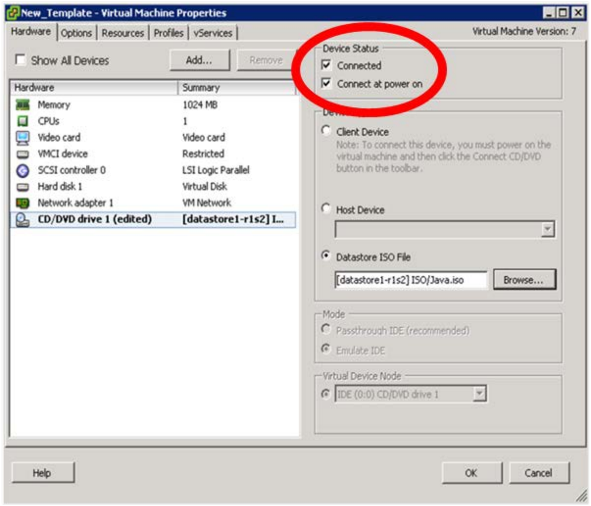


Figure 123. VM Hardware Properties Device Status Selections.

7. The ISO will now be accessible to the VM via the CD-ROM drive.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

[1] I. Foster, *et al.* (2001, September). *Cloud Computing and Grid Computing 360-Degree Compared* [Online]. Available: http://arxiv.org/ftp/arxiv/papers/0901/0901.0131.pdf

[2] NIST. (2011, September). *The NIST Definition of Cloud Computing* [Online]. Available: http://csrc.nist.gov/publications/PubsSPs.html

[3] J. B. Michael *et al*., "Leveraging the Cloud to Support Communications in the Tactival Environment," Naval Postgraduate School; Monterey, CA, Technical Report 2011 [Online]. Available: http://www.dtic.mil/dtic/tr/fulltext/u2/a555277.pdf

[4] V. Clark. (2002, October). *Sea Power 21* [Online]. Available: http://www.navy.mil/navydata/cno/proceedings.html

[5] C. Chaubal. (2008, October) *Architecture of VMware ESXi* [Online]. Available: http://www.vmware.com/files/pdf/ESXi_architecture.pdf

[6] R. Troy and M. Helmke, "VMware ESXi 5.0 Server Overview," in *VMware Cookbook*, 2nd ed. USA: O'Reilly Media, Inc., 2012, ch. 1, sec 1.4.

[7] VMware. (2012, Jun) *VMware vSphere ESX and ESXi Info Center* [Online]. Available: http://www.vmware.com/products/vsphere/esxi-and-esx/compare.html

[8] VMware. (2012) *VMware vCenter Server* [Online]. Available: http://www.vmware.com/products/vcenter-server/overview.html

[9] J. King. (2012, September) *VMware vSphere Blog.* [Online]. Available: http://blogs.vmware.com/vsphere/2012/09/vcenter-single-sign-on-part-1-what-is-vcenter-single-sign-on.html

[10] VMware. (2012, July) *Technical Marketing Documentation* [Online]. Available: "http://www.vmware.com/files/pdf/techpaper/Whats-New-VMware-vCenter-Server-51-Technical-Whitepaper.pdf

[11] T. White, *Hadoop: The Definitive Guide,* 3rd ed. USA: O'Reilly Media, Inc., 2012.

[12] B. Hedlund. (2011, September) *Studies in Data Center Networking, Virtualization, Computing* [Online]. Available: http://bradhedlund.com/2011/09/10/understanding-hadoop-clusters-and-the-network/" \l "download

[13] R. McDougall. (2012, June) *VMware: Office of the Chief Technology Office* [Online]. Available: http://cto.vmware.com/project-serengeti-theres-a-virtual-elephant-in-my-datacenter/

[14] VMware Inc. (2012, August) *Project Serengeti* [Online]. Available: http://serengeti.cloudfoundry.com/pdf/Serengeti-Users-Guide.pdf

[15] T. D. Nguyen *et al.*, "Towards a Cross-Domain MapReduce Framework," in *Military Communications Conference,* San Diego, CA, 2013, pp. 18–21

[16] VMware. (2012) *vSphere Installation and Setup, vSphere 5.1* [Online]. Available: http://pubs.vmware.com/vsphere-51/topic/com.vmware.ICbase/PDF/vsphere-esxi-vcenter-server-51-installation-setup-guide.pdf

[17] S. Lowe, *Mastering VMware Sphere 5*. Hobooken, NJ: Sybex, 2011.

[18] J. Buell, *A Benchmarking Case Study of Virtualized Hadoop Performance on VMware vSphere 5* [Online]. Available: https://www.vmware.com/files/pdf/VMW-Hadoop-Performance-vSphere5.pdf

[19] M. Noll. (2011, April) *Benchmarking and Stress Testing an Hadoop Cluster With TeraSort, TestDFSIO & Co*. [Online]. Available: http://www.michael-noll.com/blog/2011/04/09/benchmarking-and-stress-testing-an-hadoop-cluster-with-terasort-testdfsio-nnbench-mrbench/

[20] VMware. (2013) *VMware vSphere ESXi and ESX Info Center* [Online]. Available: http://www.vmware.com/products/vsphere/esxi-and-esx/management.html

[21] Microsoft. (2007, April) *Active Directory Domain Services Overview* [Online]. Available: http://technet.microsoft.com/en-us/library/cc731053(WS.10).aspx

[22] VMware Serengeti. (2012, May) *Serengeti-Pantry Serengeti Cookbooks* [Online]. Available: https://github.com/vmware-serengeti/serengeti-pantry

[23] C. Scott. (2012, October) *Big Data Analytics in the Maritime Battlespace* [Online]. Available: http://ctovision.com/2012/10/big-data-and-analysis-in-the-maritime-battlespace/

[24] VMware. (2012, June). *Virtualizing Apache Hadoop* [Online]. Available: http://www.vmware.com/files/pdf/Benefits-of-Virtualizing-Hadoop.pdf

[25] The Apache Software Foundation. (2013, July). *Apache Hadoop* [Online]. Available: http://hadoop.apache.org

[26] D. DeWitt. (2011). *Keynote Presentation* [Online]. Available: http://simranjindal.com/2011/10/17/remote-attendees-reflections-sqlpass-2011-day-3-keynote-by-dr-david-dewitt/

[27] VMware. (2012). *Company Page* [Online]. Available: http://www.vmware.com/comapny

[28] J. Hales *et al.*, *Administering vSphere 5*. Boston, MA: Course Technology PTR, 2013.

[29] S. Sorkin. (2011). *Large-Scale, Unstructured Data and Analysis usig Splunk* [Online]. Available: http://www.splunk.com/web_assets/pdfs/secure/Splunk_and_MapReduce.pdf

[30] Intel. (2012, August). *Intel's IT Manager Survey on How Organizations Are Using Big Data* [Online]. Available: http://www.intel.com/content/dam/www/public/us/en/documents/reports/data-insights-peer-research-report.pdf

[31] D. Kirkpatrick. (2011, November 30). *Now Every Company is a Software Company* [Online]. Available: http://www.forbes.com/sites/techonomy/2011/11/30/now-every-company-is-a-software-company/

[32] C. Young. (2012, March 12). *Military Intelligence Redefined Big Data in the Battlefield* [Online]. Available: http://www.forbes.com/sites/techonomy/2012/03/12/military-intelligence-redefined-big-data-in-the-battlefield/%20

[33] M. Rouse. (2012, January 10). *Big Data Analytics* [Online]. Avaialble: http://searchbusinessanalytics.techtarget.com/definition/big-data-analytics%20

[34] Department of Defense. (2012, February). *Sustaining U.S. Global Leadership: Priorities For 21st Century Defense* [Online]. Available: http://www.defense.gov/news/Defense_Strategic_Guidance.pdf

[35] J. Rittinghouse and R. James, *Cloud Computing: Implentation, Management, and Security*. Boca Raton, Florida, USA: CRC Press.

[36] J. Manyika et al. (2011, May). *Big Data: The next frontier for innovation, competition, and productivity* [Online]. Available: http://www.mckinsey.com/Insights/MGI/Research/Technology_and_Innovation/Big_data_The_next_frontier_for_innovation

[37] E. Brynjolfsson and A. McAfee. (2012, October) *Big Data: The Management Revolution* [Online]. Available: http://hbr.org/2012/10/big-data-the-management-revolution

[38] J. Dean and S. Ghemawat. (2004) *MapReduce: Simplified Data Processing on Large Clusters* [Online]. Available: http://static.googleusercontent.com/external_content/untrusted_dlcp/research.google.com/en/us/archive/mapreduce-osdi04.pdf

[39] R. Natarajan. (2010, June 2) *VMware Virtualization Fundamentals - VMware Server and VMware ESXi* [Online]. Available: http://www.thegeekstuff.com/2010/06/vmware-server-and-vmware-esxi-introduction/

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California