

Virtual Design Master

Challenge 4 –Beyond the Clouds

Timothy J. Patterson

ABSTRACT

Now that we have successfully established ourselves in our temporary Moon base and fortified what is left of the island nations on Earth, we must now focus solely on our final destination – Mars. An infrastructure must be build to support our environments. Power, space, and cooling are all very limited for this journey.

Table of Contents

1. Purpose and Overview	3
1.1. Executive Summary.....	3
1.2. Summary Analysis	3
1.3. Intended Audience.....	3
1.4. Requirements.....	3
1.4.1 Availability.....	3
1.4.2 Maintainability	3
1.4.3 Integrity	4
1.4.4 Reliability	4
1.4.5 Safety.....	4
1.4.6 Scalability.....	5
1.4.7 Automation.....	5
1.5. Constraints.....	5
1.6. Risks.....	6
1.7. Assumptions.....	6
2. Architecture Design	7
2.1. Physical Design	7
2.2. Logical Design.....	7
2.2.1 Parent Host Layout.....	7
2.2.2 vSphere Design.....	9
2.2.3 vCloud Design: Preparation	11
2.2.4 vCloud Design: Provider Datacenter	11
2.2.5 vCloud Design: External Networking.....	12
2.2.6 vCloud Design: Network Pool	12
2.2.7 vCloud Design: Organization Virtual Datacenter.....	12
2.2.8 vCloud Design: Edge Gateway.....	13
2.2.9 Reporting: vCenter Orchestrator.....	14
2.3 Virtual Application Design	14
3. Future Deployment Guidelines	14
3.1. Future Architecture.....	15
3.2. An Eye Towards the Future	15
Access Instructions / Server <-> IP Mappings.....	16

1. Purpose and Overview

1.1. Executive Summary

Now that we have successfully established ourselves in our temporary Moon base and fortified what is left of the island nations on Earth, we must now focus solely on our final destination – Mars. An infrastructure must be build to support our environments. Power, space, and cooling are all very limited for this journey. This infrastructure must be built in such a way that it will continue to be suitable once we are permanently settled on the red planet. The systems will be connected to a new environment called “The Elysium Grid”, which will be the foundation of the new Martian infrastructure.

1.2. Summary Analysis

1.3. Intended Audience

This document is aimed at the unfortunate IT professionals who drew the shortest straw from the bunch and have to administer this environment within these nearly impossible constraints.

1.4. Requirements

This design must run on a single Dell PowerEdge M610 server with 16GB of RAM, a single quad-core Xeon E5506 processor, and a single 250GB hard drive. Additionally, this design must incorporate vCenter, vCloud Director, vCenter Orchestrator, and must run entirely on virtual distributed switches. This is all done to support a critical application that includes both a Linux and a Windows component. This environment will utilize VXLAN to provide a virtualized network, as well as Docker for application layer abstraction.

1.4.1. Availability

Given the constraints, there is no way to guarantee uptime, however the design must make every effort to remain available at all times when possible.

R001	“Best effort” uptime must be a priority.
-------------	---

1.4.2. Maintainability

Maintainability must be ensured. Since we are dealing with many different components, there are many moving parts to this design. Documentation is a must have tool for ongoing system maintainability. This is especially true once our permanent home is established on Mars and the infrastructure will be interconnected with that of the other ships.

R002	Must be able to understand all of the components of the existing infrastructure.
R003	Design must be documented properly.

1.4.3 Integrity

System integrity ensures that adds / moves / changes are not done on production systems. Unfortunately for us, we have no room for error, as this environment is a single production deployment. As such, a change control board procedure is imperative. A change log should also be kept so that adverse changes can be easily reverted.

R004	A change log should be kept to ensure adverse changes can be reverted.
R005	A change control board must be established and ALL changes must be approved.

1.4.4 Reliability

Reliability of the infrastructure must be guaranteed. The weight of meeting this goal falls onto the infrastructure engineers and the local implementation teams. These people must take all of the necessary precautions to ensure the infrastructure runs without errors.

R006	A peer-based design review process needs to be put into place.
R007	Infrastructure component configurations must be audited before placing into production.

1.4.5 Safety

Since this entire infrastructure is contained within a single physical server, we must never allow anyone to power down, touch, breathe on, or even think about this physical machine. It must be protected with our lives at all costs. The only exception for this is emergency maintenance. Emergency maintenance must only take place after passing a majority vote and should be performed with the hands of our local brain surgeon.

R008	Physical security is a top priority.
R009	A brain surgeon must be among our ship's population.

1.4.6 Scalability

This design must have the ability to scale up and out once we reach our final destination on Mars. In order to achieve this, the design must be implemented with forethought and be built with common tools. Both Linux and Windows are in use powering the target applications.

R010	The design must support integrating with other deployments of its kind.
R011	The design must be able to scale out by supporting the addition of more hardware.
R012	The design must be able to scale up by adding more resources to the existing hardware.
R013	The system must support Linux and Windows on all hypervisors used.

1.4.7 Automation

In order to ensure the infrastructure is operating to the best of its ability, the design should take advantage of an automated workflow that checks and reports on key metrics. This will help us determine the health of the system as a whole.

R014	The system must use vCenter Orchestrator to generate an automatic health report.
-------------	---

1.5. Constraints

C001	The infrastructure must be deployed on the provided single physical server. Everything must run within 16GB of RAM. It is all we have folks!
C002	The infrastructure must use a nested deployment of vSphere hosts. Because nesting is a very cool and very powerful technique!
C003	Only production software can be used. Sorry, vSphere version 5.5.5 (created by the VDM team) cannot be used!
C004	The architecture must include vCenter Server. We will need it to manage this environment!
C005	The architecture must include vCloud Director. vCloud Director is the bomb-diggity!

C006	The architecture must use vCenter Orchestrator.
	Because Melissa said so.
C007	All networks must exist on Distributed Virtual Switches.
	To cause problems and make me lose two days to troubleshooting...
C008	The environment must contain at least two active, accessible guests. One must be Linux, the other Windows.
	No reason to be hatin' on one OS over another!
C009	VXLAN must be used in the host environment.
	This supports ultimate portability and will aid us in the future.
C010	Docker must be used to host the application.
	This will allow us to move our container to a similar, but different host in the future.
C011	We only have a /29 of public IP space available to us.
	Makes me wonder where all the IPv4 addresses have gone with the Earth in shambles...

1.6. Risks

I001	This entire infrastructure resides on a single physical server. This is ambitious, dangerous, challenging, and a pain in the behind all at the same time.
I002	Again, this entire infrastructure resides on a single physical server. If any part of the hardware dies, we are seriously screwed!
I003	The hardware may not be powerful enough to run the given application. I am very grateful that we have 16GB RAM to work with instead of the original 12GB!

1.7. Assumptions

A001	The available hardware is on the VMware HCL. The hardware provided to us must be supported with our hypervisor.
A002	All equipment in this design is new and validated to function properly. The equipment has run for 1 week. All tests were passed without issue.
A003	The hardware has been certified to run given our environmental constraints. The hardware has adequate cooling and power within its container.
A004	Infrastructure team will not do software development. A separate team exists to maintain the application stack and deploy it to the provided infrastructure.

2. Architecture Design

Within this section the production deployment that has been built will be described.

2.1. Physical Design

The physical design for this environment is very simple. The lab itself is built upon a single Dell PowerEdge M610 server with the following specs:

- 1 x Intel Xeon E5506 CPU
 - Running at 2.13GHz
 - 4 cores per socket
 - 4 logical processors (not hyperthreaded)
- 16GB RAM
- 230GB hard drive -- single disk, no RAID
- 1 x 1Gbps NIC
 - Connected to a public, Internet facing network
 - A /29 has been allocated for our use

2.2. Logical Design

While the physical design for this environment is simple, the logical design is very complicated. Buckle your seatbelts and hang on for the ride!

2.2.1 Parent Host Layout

The parent host (the physical server) is running a Dell-customized version of ESXi 5.5 installed onto its local disk. The remainder of the local disk has been provisioned into a single datastore named 'local-parent01'.

To enable use of the nested ESXi hosts, the appropriate CPU flags have been enabled in the BIOS and are allowed to pass into the nested hosts.

The following virtual machines exist and are running on the parent host:

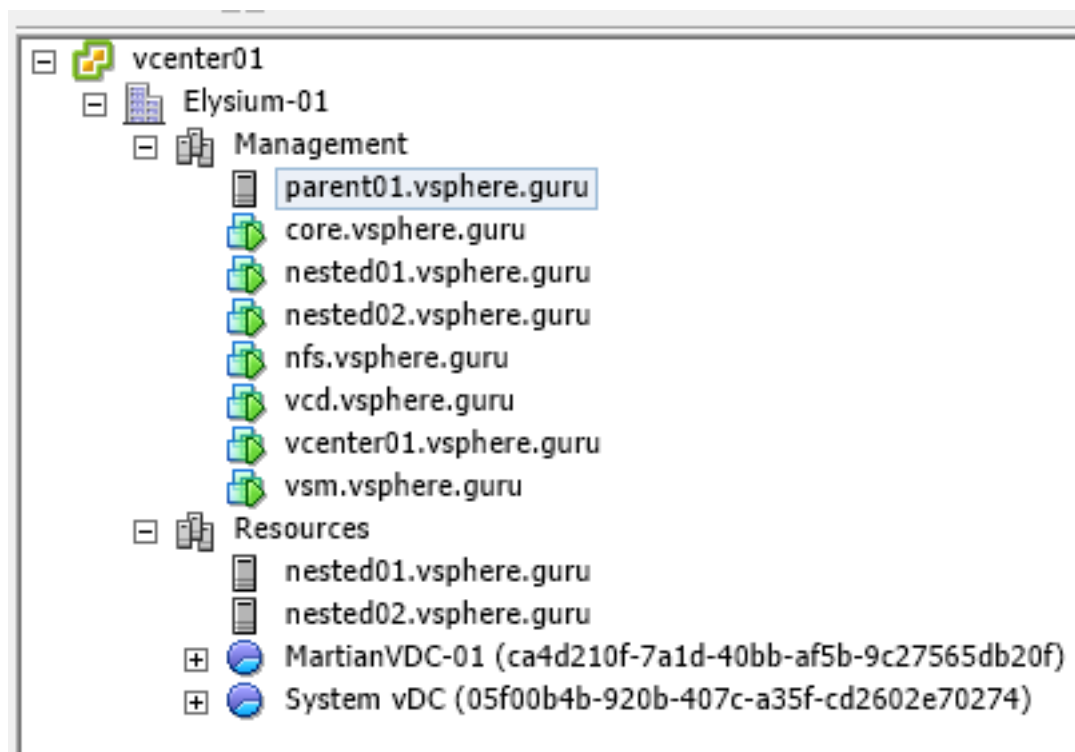
- **core.vsphere.guru**
 - Windows Server 2012 R2
 - This VM provides DHCP, DNS, NAT, and VPN access to the management network. It also home to the vCenter Orchestrator portion of the environment.
 - 2 vCPUs
 - 2 GB RAM
 - 20 GB hard disk, thin provisioned
 - Connected to the dvManagement and dvPublic networks

- **nfs.vsphere.guru**
 - CentOS 6.5 64-bit, minimal installation
 - This VM has been allocated a 100GB portion of the local-parent01 datastore. Its sole purpose is to expose this storage to the rest of the environment via the NFS protocol.
 - 1 vCPU
 - 512 MB RAM
 - 10 GB hard disk for OS, thin provisioned
 - 100 GB hard disk for shared storage, thick provisioned
 - Connected to the dvManagement network
- **vcenter01.vsphere.guru**
 - vCenter Server Appliance
 - This VM provides a single point of management for the raw VM environment and acts as the central hub to the entire infrastructure.
 - 2 vCPUs
 - 4 GB RAM
 - 130 GB hard disk, thin provisioned
 - Connected to the dvManagement network
- **vsm.vsphere.guru**
 - vShield Manager Appliance
 - This VM provides the management of the vShield infrastructure. It is responsible for the configuration and orchestration of the network virtualization (VXLAN) deployment and edge services.
 - 2 vCPUs
 - 1 GB RAM
 - 60 GB hard disk, thin provisioned
 - Connected to the dvManagement network
- **vcd.vsphere.guru**
 - vCloud Director Appliance
 - This VM powers the entire vCloud Director portion of the infrastructure. It takes advantage of the resources provided by the nested ESXi hosts and the VCSA and VSM appliances.
 - 1 vCPU
 - 2.5 GB RAM
 - 30 GB hard disk, thin provisioned
 - Connected to the dvManagement network
- **nested01.vsphere.guru**
 - Nested ESXi 5.5 host.
 - This VM is used to provide resources to the vCloud Director environment

- 2 vCPUs
 - 4 GB RAM
 - 20 GB hard disk, thin provisioned
 - Connected to the dvManagement and dvPublic networks
- **nested02.vsphere.guru**
 - Nested ESXi 5.5 host.
 - This VM is used to provide resources to the vCloud Director environment
 - 2 vCPUs
 - 4 GB RAM
 - 20 GB hard disk, thin provisioned
 - Connected to the dvManagement and dvPublic networks

2.2.2 vSphere Design

The vSphere design is relatively simple.

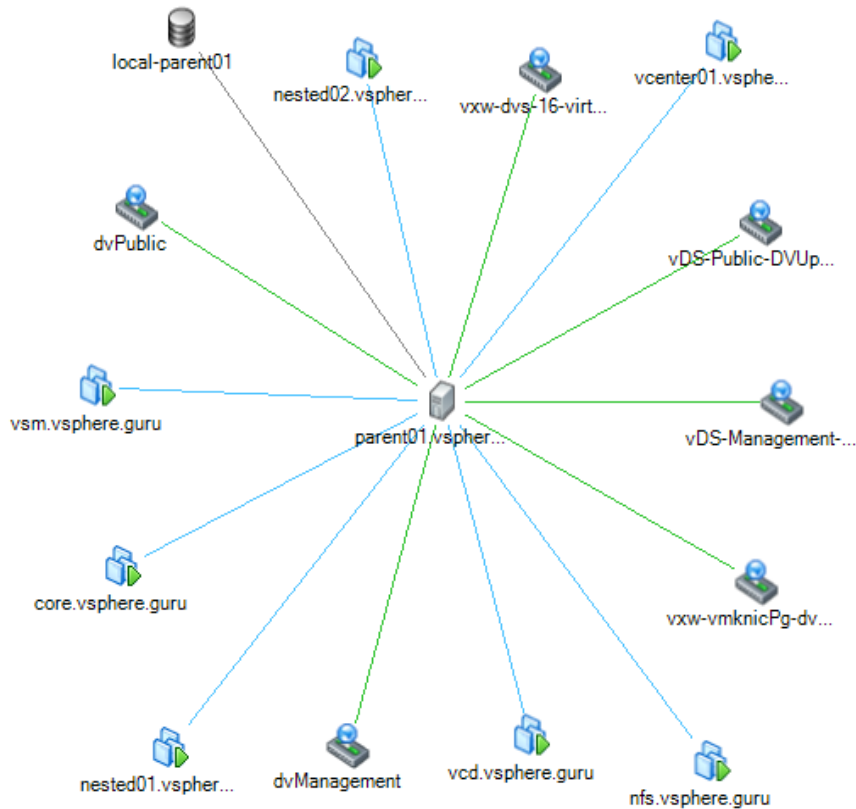


- A single datacenter object exists, Elysium-01
- There are two logical clusters:
 - **Management:** Contains all of the VMs that power the services required by the infrastructure.

- **Resources:** Contains the nested ESXi hosts that power the vApps residing within vCloud Director.
- DRS is enabled on both clusters to aid with initial placement and load balancing within the Resources cluster (set to fully automatic).
- HA **is not** used due to the lack of redundant hosts in this environment.
- Shared storage is provided by the nfs.vsphere.guru VM.
 - There is a single datastore named 'NFS-01' that is mounted by all ESXi hosts in the environment.
 - The datastore has been tagged and set up with a sample storage profile. This enables its use within vCloud Director.
- Host networking exists purely on distributed virtual switches with the parent host and the nested hosts connected to them:
 - **vDS-Management:** Provides ESXi host interconnectivity and is only for backend management purposes. No physical uplinks exist for this switch.
 - **vDS-Public:** Provides access to the public Internet network via the parent host's single physical uplink. This vDS is utilized by vShield Manager to provide virtual datacenter networking to vApps running in vCloud Director.

Name	State	VDS Status	Status
parent01.vsphere...	Connected	Up	Nor...
nested02.vsphere...	Connected	Up	Nor...
nested01.vsphere...	Connected	Up	Nor...

It is very important to understand that **all** resources spawn from the single physical host:



2.2.3 vCloud Design: Preparation

vShield Manager ('vsm') has been deployed and connected to 'vcenter01'. Additionally, the 'Resources' cluster has been through the VXLAN preparation process.

The vCloud Director appliance has been deployed and connected to both 'vcenter01' and 'vsm'.

2.2.4 vCloud Design: Provider Datacenter

vCloud Director has been configured with a single provider datacenter (PVDC), named 'Elysium01-PVDC'. This PVDC has been allocated all resources found in the 'Resources' cluster in vSphere.

This equates to a total resource pool of:

- CPU: 4.94 GHZ
- Memory: 2.55 GB
- Storage: 98 GB

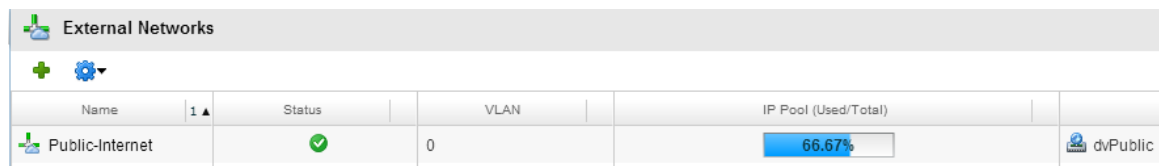
Since only ESXi 5.5 is used within the environment, the highest supported hardware version for VMs has been set to 10.

2.2.5 vCloud Design: External Networking

An external network named 'Public-Internet' has been defined. This definition connects with the 'dvPublic' portgroup (residing on the vDS-Public distributed switch).

Three IP addresses from our /29 allocation have been configured for use with this network.

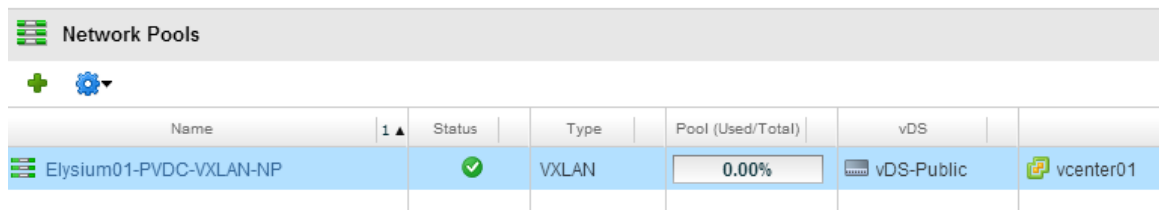
This external network will allow the VMs running in the vCloud environment to access the Internet.



External Networks					
Name	Status	VLAN	IP Pool (Used/Total)		
Public-Internet	✓	0	66.67%		dvPublic

2.2.6 vCloud Design: Network Pool

A network pool named 'Elysium01-PVDC-VXLAN-NP' was automatically created as part of the external networking provisioning process. This network pool is backed by VXLAN technology and is fully visible within the vShield Manager appliance.



Network Pools						
Name	Status	Type	Pool (Used/Total)	vDS		
Elysium01-PVDC-VXLAN-NP	✓	VXLAN	0.00%	vDS-Public		vcenter01

2.2.7 vCloud Design: Organization Virtual Datacenter

An organization named 'MartianPeeps' was created. This organization defines a single administrative user named 'martian'.

Additionally, a virtual datacenter (VDC) was provisioned for 'MartianPeeps', named 'MartianVDC-01'.

This VDC was statically allocated 2 GHz of CPU power and 1.5 GB of RAM. This leaves enough overhead in the PVDC for the necessary vShield Edge appliance deployment that will be mentioned later.

The VDC is a member of the 'Elysium01-PVDC-VXLAN-NP' network pool. A virtual subnet (10.10.10.0/24) exists for this VDC within this network pool.

A catalog has been created and assigned to this VDC. Its use in our deployment is limited to hosting virtual media ISO files.

2.2.8 vCloud Design: Edge Gateway

When the 'MartianVDC-01' virtual datacenter was provisioned and joined to its network pool, vShield Manager coordinated the deployment of a vShield Edge appliance. This appliance acts as a gateway for our virtualized network.

The appliance has been assigned the IP address of 10.10.10.1. It has been configured to provide DHCP and NAT services to the VMs residing within the VDC.

Edge Gateways				
Name	Status	Organization VDC	# External Networks	# Organization VDC Netw
Martian-VDC-Edge-01	✓	MartianVDC-01	1	1

Configure Services: Martian-VDC-Edge-01				
DHCP NAT Firewall Static Routing VPN Load Balancer				
Dynamic Host Configuration Protocol (DHCP) automates IP address assignment to virtual machines connected to organization VDC networks. You can config and manage IP address ranges and lease parameters for each of the organization VDC networks connected to this edge gateway.				
<input checked="" type="checkbox"/> Enable DHCP				
Applied On	IP Range	Default Lease	Max Lease	Enabled
Martian-VDC-Network-01	10.10.10.10-10.10.10.75	3600	7200	✓

Configure Services: Martian-VDC-Edge-01							
DHCP NAT Firewall Static Routing VPN Load Balancer							
Network Address Translation (NAT) modifies the source/destination IP addresses of packets arriving to and leaving from this Edge Gateway. Source NAT (SNAT) translates the source address of a packet before leaving this gateway, whereas Destination NAT(DNAT) translates the destination IP address/port of a packet received by this gateway.							
Applied On	Type	Original IP	Original Port	Translated IP	Translated Port	Protocol	Enabled
Public-Internet	SNAT	10.10.10.0/24	any	204.10.110.195	any	ANY	✓

In order to provide the NAT service, a SNAT rule was created. This rule ties together a source range of internal IP addresses and an IP address that has been sub-

allocated out of the public IP pool attached to our public network. Additionally, the firewall has been disabled for simplicity (in a production environment the firewall rules should be granularly configured).

2.2.9 Reporting: vCenter Orchestrator

A very simple vCenter Orchestrator workflow has been established that will check the amount of active and free memory for parent01.vsphere.guru and write the values to a file for later inspection. This is just to sample the capabilities of vCenter Orchestrator itself.

2.3 Virtual Application Design

While it is outside of the scope of this document to detail how the application works, it is fully within our realm to describe the vApp container that houses the VMs that the application will live upon.

A single vApp named 'vApp_SaveManKind' has been provisioned. It contains the following two VMs:

- Linux01:
 - 1vCPU
 - 512 MB RAM
 - 1 NIC: connected to the 'Martian-VDC-Network-01' network
 - 10 GB hard disk
- Windows01:
 - 1vCPU
 - 1 GB RAM
 - 1 NIC: connected to the 'Martian-VDC-Network-01' network
 - 10 GB hard disk

The VM 'Linux01' hosts a Docker container for our application to reside upon. For illustration purposes a simple command-line shell has been opened within a Docker container.

3. Future Deployment Guidelines

Due to the constraints presented in this environment a lot of "standard" design practices were bypassed in order to produce a working environment.

Many single points of failure exist within this design: the single physical server, the omission of the HA feature within vSphere, a single node for vCenter Server, etc.

When we finally arrive on Mars we will want to transform this environment into something much more sustainable so that our future will be preserved.

3.1. Future Architecture

While the components to production architecture were mostly all represented in this environment, the resiliency was not. In a production deployment, single points of failure cannot exist.

With that said, in the future we must eliminate all single points of failure by adding additional hardware and taking action to ensure critical servers and services are running at all times.

This infrastructure design supports both scaling up and scaling out. More powerful hardware will greatly improve overall performance within this environment, however the scale up model alone does not fix our redundancy problems.

The design can easily scale out simply by adding more hosts to the 'Resources' cluster. A fact that is perhaps more important though, this design can be consumed into a much larger vCloud deployment with little effort. Likewise, it can consume other vCloud cells.

The use of abstraction technologies like VXLAN for networking and Docker for applications enables great portability via encapsulation. The flexibility generated allows us to deploy our environment in nearly any scenario that the future may hold.

3.2. An Eye Towards the Future

The future is always a great unknown. It is my sincere hope that mankind will survive and thrive in the Martian environment. Thankfully the plague of the zombies is behind us once and for all!

As we prepare ourselves to design and maintain the infrastructure of the future it is very important that we continue to seek out the elite engineers from our population. After all, we owe our lives and our future to a very talented Virtual Design Master!

I leave you with one word: **Godspeed.**

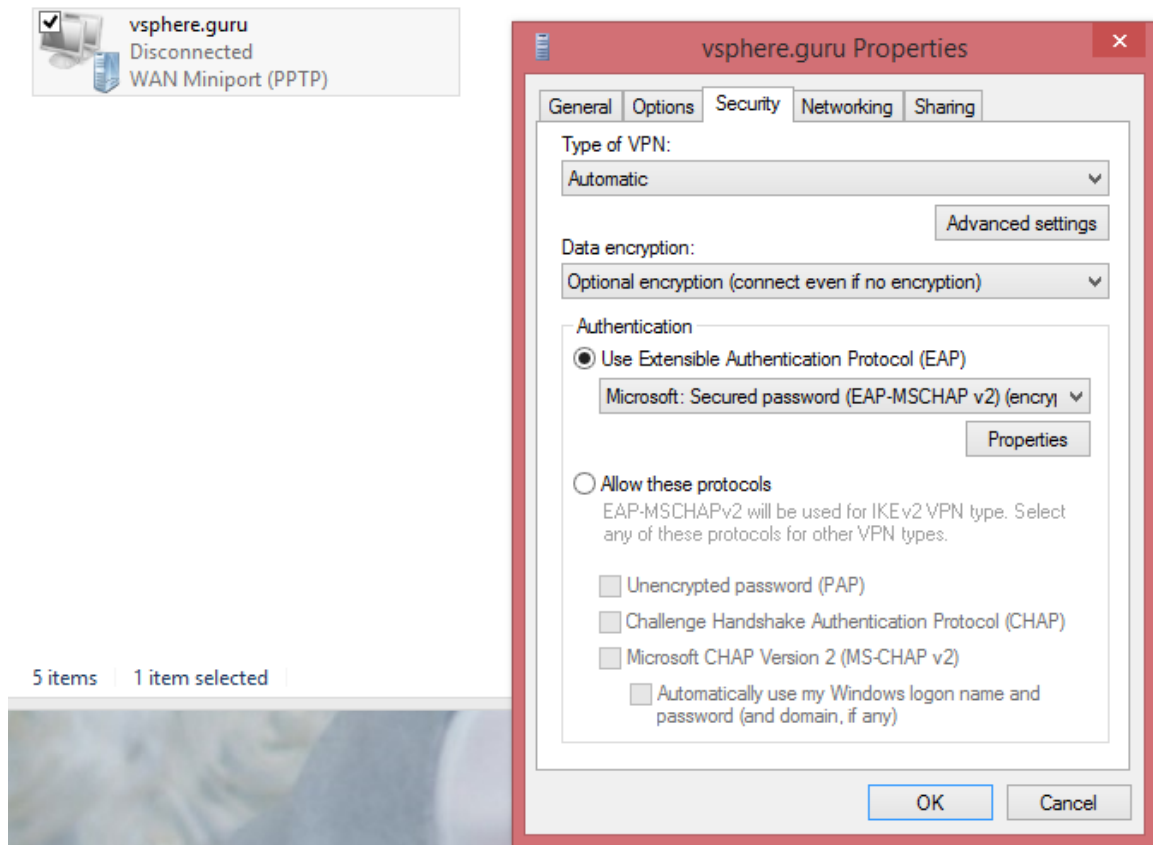
Access Instructions / Server <-> IP Mappings

The password used throughout the entire infrastructure is: G0dSp33d

There is a PPTP VPN set up that allows you to access core.vsphere.guru. It is actually resolvable in DNS from the Internet.

Use the username 'Administrator' and point your OS's VPN client to: core.vsphere.guru

(Note: This works on both Mac OS X and Windows Vista+)



Alternatively, core.vsphere.guru has RDP enabled. You can access it via the client of your choice, however, be prepared for the experience to be rather... slow.

Once you are logged in, you can access the following servers via DNS or IP:

Server / DNS:	IP:	Username:	Password:	Notes:
core	204.10.110.197 / 10.0.0.1	Administrator	G0dSp33d	RDP & PPTP VPN accessible
vcenter01	10.0.0.5	root & Administrator	G0dSp33d	
nfs	10.0.0.2	root	G0dSp33d	
parent01	204.10.110.193 / 10.0.0.10	root	G0dSp33d	
vcd	10.0.0.50	Administrator	G0dSp33d	
vsm	10.0.0.52	Admin	G0dSp33d	
nested01	10.0.0.80	root	G0dSp33d	
nested02	10.0.0.81	root	G0dSp33d	

Note: parent01 has a public, external IP. This was for “emergency” purposed only and is not mentioned throughout the design.

BareMetalCloud credentials: vdm2 / bhXQ2TAM