# Taoslab for Admins

# Current setup:

* 3 Hyper-V hosts
* 2 NetApp Filer Heads (details in Appendix)
* 3 VMware hosts
* 6 Servers for class or eval use.
* 1 vCenter running on a Hyper-V host
* 1 ASA 5510 204.107.230.112– Providing Firewall, VPN, and Routing (management on production network)
* 2 Dell Switches – providing Layer 2 switching (management on Production network)

# Internal Access:

If you are connecting from within Boise’s network, you can try your taos credentials but these may not work because they are not domain admins. It is better to use taoslab credentials. DNS names usually resolve, but remember the taoslab.local suffix.

# External Access:

We have two VPN methods. The preferred way is to use <https://taoslab.taos.com>. You must be a member of one of the below Groups.

|  |  |  |
| --- | --- | --- |
| Taos Group (**NOT** Taoslab) | Access | Notes/Comments |
| Taos Training Class and Lab space | 10.31.3.0/24 & 10.31.4.0 | Classroom section of Lab as well as Lab Test space |
| Taos Lab | 10.31.2.0/24 | CCNA Lab |
| Network Device Admins | 10.31.3.0/24, 172.31.93.0/24, 10.31.1.0/24, 10.31.2.0/24, 10.31.99.0/2 | Admin Parts of Lab |
|  | 10.31.5.0/24 | Delphix Lab |

IPSec VPN info is below.

Below is a zip with three PCFs. LAB-Admin.PCF allows access to the admin parts of the lab. TaosTraining is what most users need for Classes and test sysstems. Taos Lab User-CCNA.pcf is for the Network/CCNA class users.



Currently Boise has access to all parts of the lab directly.

|  |  |  |
| --- | --- | --- |
| Networks | Comments | Notes |
| 172.31.93.0/24 | Legacy portion of the lab that is in the prod network. | To be removed from TaosLab entirely |
| 10.31.1.0/24 |  | VPN admin users land here |
| 10.31.2.0/24 | CCNA Lab |  |
| 10.31.3.0/24 | Classroom (specifically set aside from classes being taught) | VMware vSwitch = Training |
| 10.31.4.0/24 | Non-class test systems | Longer term systems: vswitch = LabTestSystems |
| 10.31.5.0/24 | Available Lab | Available |
| 10.31.50.0/24 | VPN users (this is IP space VPN users use) | Users that don’t need access to admin sections (?) |
| 10.31.99.0/24 | Lab Infrastructure (servers, storage, etc.. things that provide infrastructure for lab) | Runs on Hyper-V cluster (3 IBM servers) |

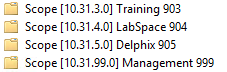
# Hyper-V Lab Infrastructure Information

* 3 Servers running Windows Server 2012 R2
* Hosts:
  + Boi-labhvh02a
  + Boi-labhvh02b
  + Boi-labhvh02c
* Guests (core infrastructure)
  + DC (to be created)
  + Monitoring (to be created)
  + WSUS (BOI-LABWSUS)
  + System Center (BOI-LABSC. Software not installed)
  + vCenter (BOI-LABVCENTER)

# VMware Lab Information

1. The vCenter (where most VMs are created) is BOI-LABVCENTER1.taoslab.local
2. You may access vCenter via
   1. the web client @ <https://boi-labvcenter.taoslab.local:9443/vsphere-client/>  
      Login to the web client by typing “taoslab\username” and your password
   2. <https://boi-labvcenter.taoslab.local/client/VMware-viclient.exe>
3. Note: The SSO/PSC password is T@os2014

# Active Directory Information

1. Normal naming convention is FirstNameLastInitial (GrantB).
2. When MS engineers are added in TaosLab, follow the [SOP in the section below](#_SOP:__Creating).
3. All MS engineers/OCAs are domain admins as of this date 7/22/2014
4. The only active AD server is BOI-LABDC00.taoslab.local which is a physical Dell box
5. Normal VMs for testing should go on the LabTestSystems. Training is reserved for Classes
6. DHCP scopes are   
   
7. If you need a static IP, follow the [process below](#_IP_Address_Tracking).
8. If you need to create an account follow the [SOP below](#_SOP:_Add_Interim)
9. You can access these networks from the Taos production network that your normal workstation is on. This allows you to use RDP, the vSphere client, Putty, etc. without using VPN first.

## IP Address Tracking

1. RDP to boi-labhvh02b.taoslab.local (taoslab domain)
2. Search in the start menu for “IP”
3. Right click and “Run as Administrator”
4. If the Subnets do not show up, then click File ->Open IP Management Database
5. Browse to C:\IPDatabase and click on the IPDB.
6. Currently VLANs listed are 999, 996, 903, 904, and 995. These equate to 10.31.99.0, 10.31.96.0, 10.31.3.0, 10.31.4.0, 10.31.5.0
7. If there is not a subnet, please feel free to create one.
8. Exit the app when finished.

## SOP: Add Interim Talent user to lab

1. Request that they fill out the [server request form](#_Server_Request_Form) and send it to [taoslabticket@taos.com](mailto:taoslabticket@taos.com)
2. Are they teaching a class?
3. Ask if anybody else needs access.
4. Determine if a taoslab account is necessary. It usually isn’t for linux users.
5. Submit e-mail to [servicedesk@taos.com](mailto:servicedesk@taos.com)  
   “Please add <username> to the “Taos Training Class VPN” Security Group in the Taos.local domain.
6. Deploy system requested via vcenter. Use a template if possible.
7. Get the IP and
8. Send info in e-mail with the following:
   1. username/password
   2. IP Address of the System/DNS name
   3. VPN information
9. Request confirmation and update ticket.

## SOP: Creating new user (Managed Services) in Taoslab

1. RDP to BOI-LABDC00.taoslab.local
2. Start->Run->dsa.msc (Active Directory Users and Computers)
3. Go to OU Boise->Users->Infrastructure
4. Click on the user “SuperUser” and right click->Copy
5. Fill out the name as appropriate
6. Fill out “User Logon Name” as FirstnameLastinitial (i.e. RyanM)
7. Use Taos2014 as the default password
8. Enable the Account (SuperUser is disabled)
9. Test!

# Appendix A: Server Request Form

# Appendix B: Hardware information

## Lab Infrastructure Servers

3 IBM System x3250 M3



## **VMware ESXi**

* Boi-labhvv01
* Boi-labhvv02
* Boi-labhvv03

## **Hypervisor (Free) Host for Brian Pyle** 10.31.99.211 root Taos2014

## NetApp Storage

Filers:

* <http://boi-labfiler01.taoslab.local> 10.31.99.5
* <http://boi-labfiler02.taoslab.local> 10.31.99.6

|  |  |
| --- | --- |
| **Model** | FAS3050 |
| **System ID** | 0101202409 |
| **Version** | 7.3.6 |
| **Volumes** | 3 Volumes |
| **Aggregates** | 1 Aggregates |
| **Disks** | 14 Disks  (2 spare, 0 failed) |
| **Status** | /na_admin/images/dfm/IcnNormal.gif  The system's global status is normal. |

* NFS, iSCSI, and snapMirror licensed
* Login as   
  root  
  Taos2014
* Must create Volume, LUN, and map lun to iGroup (initiator group)
* Targets:
  + 10.31.96.203 and 10.31.96.202
  + 10.31.96.201 and 10.31.96.200
* 

# Appendix C: Powershell scripts used for a Windows 8 class.

**1. Using PowerCLI (Powershell cmdlets for VMware)**

*This starts I at 1 and creates 25 VMs from templates*

*$i = 1 Do {*

*$i++*

*New-VM -vmhost boi-labhvv01.taoslab.local -Name "BOI-LABWIN81-$i" -Template Windows8.1 -OSCustomizationSpec TaosLab -Datastore na02\_General\_vmware -RunAsync }*

*While ($i -le 25)*

**2. This gets all VMs of a certain name and connects their network adapters. This is necessary because for some reason, when deploying systems from templates they start with their NICs disconnected.**

get-vm | where-object {$\_.name -like "BOI-LABWIN81-2\*"} | Get-NetworkAdapter | Set-NetworkAdapter -StartConnected $true

**3. Using the tool rdp.exe from** [**http://www.donkz.nl/**](http://www.donkz.nl/) **You can rdp into multiple machines w/ username & pw**

for /l %%i in (1,1,25) do rdp /v:boi-labwin81-%%i.taoslab.local /u:taoslab\labuser%%i /p:Lab2013

# Appendix D: Below are some Python Scripts (by Allen Smith)

I have a bunch of dhcp reservations I wanted to make for the 30 system grant spun up for the python class.

So I did the following to collect a file of the mac addresses:

$i = 1

Do {

    Get-VM pyclass-$i | Get-NetworkAdapter | `

    Select-Object -Property Parent,Name,Type,NetworkName,MacAddress,WakeOnLanEnabled | `

    Format-Table -AutoSize

$i++

}

While ($i -le 30)

I just ran that and output into a file, which I copied and pasted onto a linux box into a file called foo.

I then did the following to create a csv file I can hand to powershell:

convert.sh:

echo "ScopeId,IPAddress,Name,ClientId,Description"

i=1

cat foo |grep pyclass | awk '{ print $1, $7 }' | tr : - | while read name clientid

do

        j=$(printf '%02d' $i)

        uname=$(echo $name | tr a-z A-Z)

        echo "10.31.3.0,10.31.3.2$j,$name,$clientid,Reserved for $uname"

        let i=i+1

done

which output:

ScopeId,IPAddress,Name,ClientId,Description

10.31.3.0,10.31.3.201,pyclass-1,00-50-56-95-29-4b,Reserved for PYCLASS-1

10.31.3.0,10.31.3.202,pyclass-2,00-50-56-95-46-28,Reserved for PYCLASS-2

10.31.3.0,10.31.3.203,pyclass-3,00-0c-29-86-6b-62,Reserved for PYCLASS-3

10.31.3.0,10.31.3.204,pyclass-4,00-50-56-95-4e-3d,Reserved for PYCLASS-4

10.31.3.0,10.31.3.205,pyclass-5,00-50-56-95-37-21,Reserved for PYCLASS-5

10.31.3.0,10.31.3.206,pyclass-6,00-50-56-95-29-03,Reserved for PYCLASS-6

10.31.3.0,10.31.3.207,pyclass-7,00-50-56-95-01-c1,Reserved for PYCLASS-7

10.31.3.0,10.31.3.208,pyclass-8,00-50-56-95-2f-94,Reserved for PYCLASS-8

10.31.3.0,10.31.3.209,pyclass-9,00-50-56-95-4d-ee,Reserved for PYCLASS-9

10.31.3.0,10.31.3.210,pyclass-10,00-50-56-95-0e-03,Reserved for PYCLASS-10

10.31.3.0,10.31.3.211,pyclass-11,00-50-56-95-66-14,Reserved for PYCLASS-11

10.31.3.0,10.31.3.212,pyclass-12,00-50-56-95-15-f1,Reserved for PYCLASS-12

10.31.3.0,10.31.3.213,pyclass-13,00-50-56-95-54-a4,Reserved for PYCLASS-13

10.31.3.0,10.31.3.214,pyclass-14,00-50-56-95-7a-95,Reserved for PYCLASS-14

10.31.3.0,10.31.3.215,pyclass-15,00-50-56-95-79-df,Reserved for PYCLASS-15

10.31.3.0,10.31.3.216,pyclass-16,00-50-56-95-4c-5f,Reserved for PYCLASS-16

10.31.3.0,10.31.3.217,pyclass-17,00-50-56-95-23-2a,Reserved for PYCLASS-17

10.31.3.0,10.31.3.218,pyclass-18,00-50-56-95-13-a7,Reserved for PYCLASS-18

10.31.3.0,10.31.3.219,pyclass-19,00-50-56-95-2d-41,Reserved for PYCLASS-19

10.31.3.0,10.31.3.220,pyclass-20,00-50-56-95-32-b6,Reserved for PYCLASS-20

10.31.3.0,10.31.3.221,pyclass-21,00-50-56-95-53-54,Reserved for PYCLASS-21

10.31.3.0,10.31.3.222,pyclass-22,00-50-56-95-2f-db,Reserved for PYCLASS-22

10.31.3.0,10.31.3.223,pyclass-23,00-50-56-95-1b-6f,Reserved for PYCLASS-23

10.31.3.0,10.31.3.224,pyclass-24,00-50-56-95-66-33,Reserved for PYCLASS-24

10.31.3.0,10.31.3.225,pyclass-25,00-50-56-95-77-3d,Reserved for PYCLASS-25

10.31.3.0,10.31.3.226,pyclass-26,00-50-56-95-18-92,Reserved for PYCLASS-26

10.31.3.0,10.31.3.227,pyclass-27,00-50-56-95-4f-30,Reserved for PYCLASS-27

10.31.3.0,10.31.3.228,pyclass-28,00-50-56-95-05-3a,Reserved for PYCLASS-28

10.31.3.0,10.31.3.229,pyclass-29,00-50-56-95-70-a5,Reserved for PYCLASS-29

10.31.3.0,10.31.3.230,pyclass-30,00-50-56-95-3a-2b,Reserved for PYCLASS-30

The on the DC, from an administrative powershell:

import-csv -path .\dhcp.csv | Add-DhcpServerv4Reservation -Computername boi-labdc00

Done.

This could probably all be done in powershell.

# Appendix E: Taoslab.com information

**From:** Boris Mikhitan   
**Sent:** Tuesday, June 16, 2015 4:39 PM  
**To:** Grant Ballard; Eric Geyer; Joel Huffman; Douglas Schneider; Mike Lee; Greg Bergeson; Anthony Fuller; Roscoe Howard  
**Subject:** taoslab.com question

As per Amy – she never heard of taoslab.com and had no objections to you guys using it.

Credentials-wise  – site is hosted by DNS made easy ->  <https://cp.dnsmadeeasy.com/login> and credentials are in PMP

Here is what I see when I login and open taoslab.com entry:

