



[vSphere 4 Available for Download](#) • [Strange Error with Firefly Media Server, iTunes, and Avahi](#) »

That's it! Now, if only the rest of the process was this easy...

By the way, this same area is also where you can enable Cisco Discovery Protocol support for the dvSwitch, as I pointed out in [this recent article](#).

CREATING THE DVPORT GROUP

Like setting the MTU on the dvSwitch, this process is pretty straightforward and easily accomplished using the vSphere Client:

1. In the vSphere Client, navigate to the Networking inventory view (select View > Inventory > Networking from the menu).
2. Right-click on the dvSwitch and select New Port Group.
3. Set the name of the new dvPort group.
4. Set the number of ports for the new dvPort group.
5. In the vast majority of instances, you'll want to set VLAN Type to VLAN and then set the VLAN ID accordingly (This is the same as setting the VLAN ID for a port group on a vSwitch.)
6. Click Next.
7. Click Finish.

See? I told you it was pretty straightforward. Now on to the final step which, unfortunately, won't be quite so straightforward or easy.

CREATING A VMKERNEL PORT WITH JUMBO FRAMES

Now things get a bit more interesting. As of the GA code, the vSphere Client UI still does not expose an MTU setting for VMkernel ports, so we are still relegated to using the `esxcfg-vswitch` command (or the `vicfg-vswitch` command in the vSphere Management Assistant—or vMA—if you are using ESXi). The wrinkle comes in the fact that we want to create a VMkernel port attached to a dvPort ID, which is a bit more complicated than simply creating a VMkernel port attached to a local vSwitch.

Disclaimer: There may be an easier way than the process I describe here. If there is, please feel free to post it in the comments or shoot me an e-mail.

First, you'll need to prepare yourself. Open the vSphere Client and navigate to the Hosts and Clusters inventory view. At the same time, open an SSH session to one of the hosts you'll be configuring, and use "su -" to assume root privileges. (You're not logging in remotely as root, are you?) If you are using ESXi, then obviously you'd want to open a session to your vMA and be prepared to run the commands there. I'll assume you're working with ESX.

This is a two-step process. You'll need to repeat this process for each VMkernel port that you want to create with jumbo frame support.

Here are the steps to create a jumbo frames-enabled VMkernel port:

1. Select the host and go to the Configuration tab.
2. Select Networking and change the view to Distributed Virtual Switch.
3. Click the Manage Virtual Adapters link.
4. In the Manage Virtual Adapters dialog box, click the Add link.
5. Select New Virtual Adapter, then click Next.
6. Select VMkernel, then click Next.
7. Select the appropriate port group, then click Next.
8. Provide the appropriate IP addressing information and click Next when you are finished.
9. Click Finish. This returns you to the Manage Virtual Adapters dialog box.

From this point on you'll go the rest of the way from the command line. However, leave the Manage Virtual Adapters dialog box open and the vSphere Client running.

To finish the process from the command line:

1. Type the following command (that's a lowercase L) to show the current virtual switching configuration:
`esxcfg-vswitch -l`
At the bottom of the listing you will see the dvPort IDs listed. Make a note of the dvPort ID for the VMkernel port you just created using the vSphere Client. It will be a larger number, like 266 or 139.
2. Delete the VMkernel port you just created:
`esxcfg-vmknics -d -s <dvSwitch Name> -v <dvPort ID>`
3. Recreate the VMkernel port and attach it to the very same dvPort ID:
`esxcfg-vmknics -a -i <IP addr> -n <Mask> -m 9000 -s <dvSwitch Name> -v <dvPort ID>`
4. Use the `esxcfg-vswitch` command again to verify that a new VMkernel port has been created and attached to the same dvPort ID as the original VMkernel port.

At this point, you can go back into the vSphere Client and enable the VMkernel port for VMotion or FT logging. I've tested jumbo frames using VMotion and everything is fine; I haven't tested FT logging with jumbo frames as I don't have FT-compatible CPUs. (Anyone care to donate some?)

As I mentioned in [yesterday's Twitter post](#), I haven't conducted any objective performance tests yet, so don't ask. I can say that NFS *feels* faster with jumbo frames than without, but that's purely subjective.

Let me know if you have any questions or if anyone finds a faster or easier way to accomplish this task.

UPDATE: I've updated the comments to delete and recreate the VMkernel port per the comments below.

Tags: [CLI](#), [ESX](#), [ESXi](#), [iSCSI](#), [Networking](#), [NFS](#), [Storage](#), [Virtualization](#), [VMware](#), [vSphere](#)

38 comments

Comments feed for this article 

Trackback link: <http://blog.scottlowe.org/2009/05/21/vmware-vsphere-vds-vmkernel-ports-and-jumbo-frames/trackback/>

William on Thursday, May 21, 2009 at 2:28 pm



For Step #3 when re-creating VMkernel port on the same dvPort ID, you should be able to use the vCLI you can enable VMotion using `esxcfg-vmknics`, sadly this was not added to the Service Console version of `esxcfg-vmknics`

`--enable-vmotion`

`-E`

Enable VMotion for the VMkernel NIC on a specified portgroup.

Great read as always!

slowe on Thursday, May 21, 2009 at 2:42 pm



Ah, I see what you are saying. The `vicfg-vmknics` command found in the vSphere Management Assistant (VMA) does support an option to enable VMotion. Excellent catch, William—thanks!

Vaughn on Wednesday, May 27, 2009 at 9:34 am



Any plans to kick the tires with ALUA and the RR PSP?

slowe on Wednesday, May 27, 2009 at 9:48 am



Hey Vaughn, I'd love to kick the tires with ALUA and the RR PSP...but my NetApp is only a single-controller model. My office is right around the corner from yours...want to drop off a clustered FAS I can use for that testing? 😊

rianard on Thursday, May 28, 2009 at 1:33 pm



I have been trying this for a day and about to go crazy, am I reading this wrong, my `vmk0` is 105?
Thanks for any help...

```
[root@useresx2 ~]# esxcfg-vswitch -l
DVS Name Num Ports Used Ports Configured Ports Uplinks
dvUsersSwitch 256 2 256 vmnic2
```

```
DVPort ID In Use Client
261 1 vmnic2
262 0
263 0
264 0
133 0
134 0
```

```
DVS Name Num Ports Used Ports Configured Ports Uplinks
dvConsole 256 3 256 vmnic0
```

```
DVPort ID In Use Client
1 1 vmnic0
2 0
3 0
4 0
100 1 vswif1
```

```
DVS Name Num Ports Used Ports Configured Ports Uplinks
dvKernel 256 3 256 vmnic1
```

```
DVPort ID In Use Client
133 1 vmnic1
134 0
135 0
136 0
105 1 vmk0
```

[root@useresx2 ~]# esxcfg-vmknic -d 105
Invalid portgroup: 105

slowe on Thursday, May 28, 2009 at 2:19 pm



Hmmm...I may have a typo, there. I'll need to go back and double-check. Give me a day or two and I'll walk back through the whole configuration again to see if I misstated anything along the way.

Have you tried "esxcfg-vmknic -d vmk0"?

rianard on Thursday, May 28, 2009 at 4:49 pm



thanks for looking at it, this ended up working
esxcfg-vmknic -d -s 'dvKernel' -v 105
esxcfg-vmknic -a -i 10.10.2.16 -n 255.255.255.0 -m 9000 -s 'dvKernel' -v 105

thanks for the great blog!

joeym82956 on Thursday, May 28, 2009 at 6:17 pm



I'm having the same problem Invalid portgroup: xxx

joeym82956 on Thursday, May 28, 2009 at 7:49 pm



That worked for me

joeym82956 on Thursday, May 28, 2009 at 8:31 pm



I mean here is what worked for me

./esxcfg-vmknic -d vmk0 -v 105 -s dvKernel

fduranti on Thursday, June 4, 2009 at 2:23 pm



There's some way to create the "port" with the esxcfg-vswitch command or it's a must to create first the vmkernel port on the viciant and then delete/recreate the vmknic interface with the 9k mtu?
I've tried every flag in the esxcfg-vmknic/esxcfg-vmkswitch but it seems that I cannot connect the port to the host to create the interface automatically. My problem is that I've to create 3vmk x 12 host and it will be a bit long to create them manually and then recreate them with the command...
Anyone can help?

Pingback from [Welcome to vSphere-land! » Networking Links](#) on Thursday, June 4, 2009 at 6:34 pm

dbrowder on Friday, June 5, 2009 at 4:52 pm



Great information, thanks for sharing the tests and results here. Quick question. I'm in an environment still running V 3.5 and we are considering using jumbo frames for our unixboxes to talk to our NAS filer but will not use jumbo frames when it comes to the VMware cluster talking to the same filer. Are there any gotchas or areas to watch out for if jumbo frames are enabled on the filer but not the VM cluster?

Shudong Zhou on Tuesday, June 9, 2009 at 9:04 pm



For some reason, parts of the command was cut
esxcfg-vmknic -m 9000 -v dvPortID -s dvsName

Jim on Thursday, June 18, 2009 at 2:59 pm



Quick question. We configured a Distributed vSwitch with MTU 9000. I created a vmKernel port of a brand new host using the vSphere client, applied an IP address and saved. From the console of the host, I was able to use this command:

vmkping -s 9000

The result was a success at packet size of 9000. I did not need to remove the vmknic and re-create it from the command line.

This tells me that all created vmknics inherit the MTU setting from the switch. Could that be?

Jim on Thursday, June 18, 2009 at 3:22 pm



After a bit more experimentation, I found that you do need to re-create the vmknic with mtu 9000.

slowe on Thursday, June 18, 2009 at 3:37 pm



That was the behavior I expected! As far as I know, VMkernel NICs do *NOT* inherit the MTU from the vSwitch/dvSwitch. The output of "esxcfg-vmknic -l" should show you the MTU for each VMkernel NIC.

James S on Monday, June 29, 2009 at 8:47 pm



Another quick question... Why wouldn't you also need to modify the MTU setting on the uplink for the vSwitch where one's vmkernel is attached to??? I mean... If I have a vmkernel port with a MTU of 9000 attached to a vSwitch with a MTU of 9000 attached to an uplink with a MTU setting of 1500... isn't that a problem?

slowe on Monday, June 29, 2009 at 9:20 pm



James S, that's correct—which is why I included the section titled "Physical Network Configuration." 😊

James S on Monday, June 29, 2009 at 9:29 pm



Understood Scott...but I guess I'm still confused...configuring MTU settings on your physical switch is one thing...but let's take this example:

Your iSCSI vmkernel port is attached to vSwitch1 with a physical uplink of vmnic2...

If you check with esxcfg-vmknic -l...you note you have an MTU of 9000...great!

Then...you check your vSwitch with a esxcfg-vswitch -l and verify that it also has an MTU of 9000...wonderful...

Then...if you look at the results from an ifconfig at the service console...you see that the vmnic2 is set for 1500...not good... Or...are you saying that simply configuring the physical switch port that your uplink is attached to good enough?

Is that more clear...I'm sure I'm missing something...it happens often with me 😊

slowe on Monday, June 29, 2009 at 9:42 pm



James S,

Perhaps this will clear things up...when you use esxcfg-vswitch to set the MTU of the vSwitch to 9000, it will also set the MTU size on the linked NICs to 9000 as well. You can confirm this behavior using esxcfg-nics; you'll see that all NICs linked to the vSwitch now have their MTU set to 9000 as well.

So, now you have all the components set:

- Physical switch configured for jumbo frames
- NICs configured for jumbo frames (configured when you used esxcfg-vswitch)
- vSwitch configured for jumbo frames
- VMkernel NIC configured for jumbo frames (configured when you used esxcfg-vmknic to create the VMkernel NIC)

Does that help at all?

Marek on Wednesday, July 15, 2009 at 4:32 pm



Scott,

Do I get it right - I have almost all ESX hosts configured with MTU 9000. I would also configure 3 other servers, that have default 1500MTU. So if on these 3 servers I will do command esxcfg-vswitch -m 9000 (on every esx host) then these 3 servers will start using 9000 MTU? or should they be rebooted after this command?

slowe on Wednesday, July 15, 2009 at 5:17 pm



Marek,

Using the "esxcfg-vswitch -m 9000 vSwitchX" command will enable jumbo frames on that vSwitch, but it will not reconfigure existing VMkernel interfaces or VM interfaces to actually use jumbo frames. You'll need to re-create VMkernel interfaces and/or configure your guest operating systems separately.

Hope this helps!

Mark on Wednesday, July 29, 2009 at 5:47 pm



What about for vsphere essentials plus? We don't have vDS. Can we still use jumbo frames?

slowe on Wednesday, July 29, 2009 at 10:44 pm



Yes, absolutely. Follow these instructions:



That should take care of you!

Bozo Popovic on [Saturday, January 23, 2010 at 2:42 pm](#)



Hey Scott, great post as usual. Just to mention so nobody else gets confused. Setting vswitch and vmkernel MTU to 9000, and forgetting to configure external switches, or core switches used by the vmkernel gateway, results in Vmotion failure. I have tested these and it is quite a bad thing which by my opinion would have to be addressed by VMware soon.

I mean, could a next generation of vmknix be smart enough to send smaller packets (1500) if 9000 b large packets get rejected from the outside world (CISCO in my lab).

The errors I kept getting is that Vmotion was not possible, a general system error appeared and a Timeout was the final resolution.

Bozo

Bozo Popovic on [Monday, January 25, 2010 at 9:44 am](#)



Hello again,

one more interesting situation. I have tested Vmotion mechanism in vSphere with HP bl280g6 blades connecting to cisco 3020 lan switches each connected via 4 etherchannel uplink to 2 interconnected separate 6500 core switches. The Vmotion mechanism fails when I change uplink order from single vSwitch with two vmnics (General system error, Timeout). When I change back to the original vmnic0, vmnic1 from CLI it works fine.

Is there any resolution with further configuring CISCO networking or ESX advanced settings?

Bozo

Bozo Popovic on [Saturday, January 30, 2010 at 6:37 pm](#)



I've got all over this problem, even though VMware L2 support thinks these Vmotion + jumbo frames problems are related to some configuration on the trunked lines???

The default setting of the vSwitch was the one that should have been changed. The Failback in this configuration should have to be set to NO, otherwise if some event on the LAN network happens you might lose your hosts ability to Vmotion.

It might some time in similar environments,

Greetings,
B.

Mikhail on [Monday, February 8, 2010 at 2:49 am](#)



It's interesting, but VMkernel NIC creating on dvSwitch don't work against ESX and ESXi through vSphere CLI.

```
esxcfg-vmknix -a -i -n -m 9000 -s -v
```

"Can not specify dvsName, dvportId parameters for --add operation."

<http://communities.vmware.com/message/1472107>

Trackback from [uberVU - social comments](#) on [Monday, February 8, 2010 at 5:30 pm](#)

Karl on [Friday, March 12, 2010 at 11:20 am](#)



Great Blog!

Has anyone been through this exercise using ESXi? I am trying to drop the VMkernel port and can't quite get the syntax right. My distributed vSwitch is named dvSwitch02-iSCSI and the VMkernel port is name vmk1 with a dvport ID of 102. here is the command that I tried (with variations on the theme):

```
C:\Program Files (x86)\VMware\VMware vSphere CLI\bin>vicfg-vswitch.pl --server --userna  
me --password -d dvSwitch02-iSCSI -v 102
```

Any help would be great!

aenagy on [Thursday, March 18, 2010 at 5:59 pm](#)



Noob question:

Is it possible to configure the vSwitch for jumbo frames, but configure individual port groups to not use jumbo frames?

I ask because our VMware PSO engagement is recommending that we combine

Management and VMotion in the same vSwitch/NIC team but on separate port groups. Within each port group a different vmnic in the team/vSwitch would be configured as active and the other vmnic as standby. Like this:

```
vSwitch0
PortGroup1: Management
vmnic0 : active
vmnic1 : standby
VLAN : x
PortGroup2: VMotion
vmnic0 : standby
vmnic1 : active
VLAN : y
```

We don't want jumbo frames for the 'Management' port group as these packets need to be routable and the rest of our network will not be configured to support jumbo frames. We do want jumbo frames for the 'VMotion' port group as these packets will never leave the pSwitches and will be configured to support jumbo frames.

aenagy on [Thursday, March 18, 2010 at 6:01 pm](#)

... forgot to mention that this question is for ESXi 4.0.0 Update 1.



aenagy on [Sunday, March 21, 2010 at 2:36 pm](#)

I was in a hurry so I opened a case with VMware technical support. The short answer is no, you can't configure port groups in the same vSwitch with different MTU values.



slowe on [Sunday, March 21, 2010 at 2:40 pm](#)

Aenagy,

All port groups might have to have the same MTU, but you should be able to configure the MTU on a per-VMkernel port basis (and possibly a per-Service Console connection, too).

I'll look into that...



Tom Miller on [Monday, March 22, 2010 at 9:35 pm](#)

Scott,

The command above will not work for ESXi infrastructures:

```
3 # Recreate the VMkernel port and attach it to the very same dvPort ID:
esxcfg-vmknics -a -i -n -m 9000 -s -v
```

Please see forum article for workaround, it's clugy, but it works. Thanks to MarkEwert for the advice in the article and thanks to you for all your great post. Here is the forum article: <http://communities.vmware.com/thread/254623>



padra1g on [Wednesday, April 14, 2010 at 8:53 am](#)

if i change my vm client eth0 mtu to 9000 i cannot ftp/scp to work. However ssh/ping are ok.

To get ftp/scp to work i need to set mtu to 1500 any suggestions?

thanks,
p.



Vinod on [Tuesday, May 11, 2010 at 12:42 pm](#)

Scott,

I have two NICs. One is sharing Service console and VMkernel Port. The other is used for Virtual Machine Port. In this scenario can we still use Jumbo Frames. If yes, how would service console behave?



Reply

Name (required)

Email (required)

Website

Your comment

Submit Comment

SITE ADMIN

[Log in](#)

[Entries RSS](#)

[Comments RSS](#)

[WordPress.org](#)



Scott's Weblog by [Scott Lowe](#) is licensed under a [Creative Commons Attribution-ShareAlike 3.0 United States License](#).

 [Subscribe to feed](#)

Powered by [WordPress](#) and [Tarski](#)

5