ESXi-Arm Fling Doc

1. Downloading ESXi-Arm Fling
   1.1. Useful References
2. Supported platforms
3. I/O options
   3.1. Supported storage
      3.1.1. Disk size
   3.2. Supported networking
   3.3. USB devices
      3.3.1. Storage
      3.3.2. Networking
      3.3.3. Keyboards
   3.4. Supported plug-in PCIe adapters
4. Preparation
   4.1. iSCSI
      4.1.1. Sorting out your LUNs.
      4.1.1.1. QNAP
      4.1.1.2. Synology
      4.1.1.3. Others
      4.1.2. Configuring firmware
   4.1.3. What could go wrong?
5. Create ESXi Installer USB Key
   5.1. Using balenaEtcher on Linux, Mac or Windows
   5.2. On macOS
6. Installing ESXi-Arm
   6.1. Basic installation
   6.2. Operation with a video console
   6.3. Headless operation
   6.4. After installation
      6.4.1. Changing NIC and TCP/IP configuration
      6.4.2. Enabling SSH or console shell
7. VMware ESXi Host Client
   7.1. Login and host configuration
   7.2. Virtual Machine Creation
   7.3. Guest OS support
   7.4. See also
8. VMware vCenter Server Basics
   8.1. 1. Add a Datacenter and a Cluster
      8.1.1. 1.1. Adding a Datacenter
      8.1.2. 1.2. Adding a Cluster
   8.2. 2. Add an ESXi-Arm Fling host to vCenter Server
   8.3. 3. Create a Virtual Machine
   8.4. 4. Access VM Remote Console
9. Enabling vMotion
   9.1. Tested Platforms
   9.2. Pre-requisites
   9.3. VM migration with vMotion
10. Enabling vSphere HA
   10.1. Tested Platforms
   10.2. Pre-requisites
11. Enabling vSphere Fault Tolerance
   11.1. Tested platforms
   11.2. Pre-requisites
1. Downloading ESXi-Arm Fling

To access ESXi-Arm Fling, you’ll need to register for a MyVMware account. Registration is free - https://my.vmware.com/web/vmware/registration

To download ESXi-Arm Fling and Documentation, please head to https://flings.vmware.com/esxi-arm-edition

1.1. Useful References

- William Lam's Raspberry Pi BOM for ESXi-Arm Fling
- Cyprien Laplace’s BOM for ESXi-Arm Fling
- How to copy the Raspberry Pi UEFI files and boot ESXi-Arm using just USB device?
- Install ESXi-Arm with iSCSI
- vSAN Witness using Raspberry Pi 4 & ESXi-Arm Fling
- ESXi-Arm Fling as a lightweight vSphere Automation environment for PowerCLI and Terraform
- Installing VMware Tools on Photon OS for ESXi-Arm
- How to run Raspberry Pi OS as a VM on ESXi-Arm
- Installing VMware Tools on Raspberry Pi OS for ESXi-Arm
- Kubernetes on ESXi-Arm using k3s
- PhotonOS Arm NFS Virtual Appliance using OVF properties for ESXi-Arm
- Install VMware Tools on FreeBSD

2. Supported platforms

The Fling is launched with four platforms across a wide range of foot prints and use cases, spanning from servers and datacenters to single-board computers and far edge use cases.

- Ampere Computing eMAG 8180-based servers
- Arm Neoverse N1 Software Development Platform
- NXP LayerScape 1046A-based FRWY (4GB only)
- NXP LayerScape 1046A-based RDB
- Raspberry Pi 4B (4GB and 8GB only)

3. I/O options

3.1. Supported storage

iSCSI LUNs, NVMe and SATA drives are supported, as is USB storage. On some platforms, like the Raspberry Pi 4B, USB storage and iSCSI are the only options.

3.1.1. Disk size
The ESXi-Arm Fling is based on ESXi 7.0 and thus has a new partition scheme which consolidates VMware Tools Locker, Core Dump and Scratch partitions into a new ESX-OSData volume (based on VMFS-L). By default, the ESXi installer is expecting a disk that is 128GiB or larger.

If you insist on using a disk that is smaller than or equal to 128GiB, be sure to pass `autoPartitionOSDataSize` when the installer boot screen prompts for options (Shift-O).

E.g. `autoPartitionOSDataSize=8192` for an 8GB VMFS-L partition, and the remainder space will be used to create VMFS datastore.

For more details on changing the default OSData volume, please see this blog post.

3.2. Supported networking

PCIe and USB networking is supported. On the Raspberry Pi 4B, the onboard NIC is also supported.

3.3. USB devices

The list below is not exhaustive, but are some storage/networking options that has been tested to work with the Raspberry Pi 4B.

### 3.3.1. Storage

<table>
<thead>
<tr>
<th>VID</th>
<th>PID</th>
<th>Description</th>
<th>Product</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>154b</td>
<td>1009</td>
<td>PNY Elite 240GB USB 3.0 Portable Solid State Drive</td>
<td><a href="https://smile.amazon.com/gp/product/B01GQPXBQC">https://smile.amazon.com/gp/product/B01GQPXBQC</a></td>
<td></td>
</tr>
<tr>
<td>1f75</td>
<td>0621</td>
<td>Innostor Technology Corporation SATA Bridge/Enclosure</td>
<td><a href="https://www.amazon.com/gp/product/B07CVXS2L8">https://www.amazon.com/gp/product/B07CVXS2L8</a></td>
<td></td>
</tr>
<tr>
<td>152d</td>
<td>0583</td>
<td>JMicron NVMe Bridge/Enclosure</td>
<td><a href="https://www.amazon.com/gp/product/B07HCPCMKN">https://www.amazon.com/gp/product/B07HCPCMKN</a></td>
<td>Must be used with a powered USB hub on the Raspberry Pi 4B</td>
</tr>
<tr>
<td>090c</td>
<td>1000</td>
<td>Samsung MUF-256AB</td>
<td><a href="https://www.amazon.com/gp/product/B07D7Q41PM">https://www.amazon.com/gp/product/B07D7Q41PM</a></td>
<td>(So far) Only tested with a powered USB hub</td>
</tr>
<tr>
<td>0930</td>
<td>6545</td>
<td>Toshiba TransMemory</td>
<td><a href="https://www.amazon.com/gp/product/B00BBD7NFU">https://www.amazon.com/gp/product/B00BBD7NFU</a></td>
<td>(So far) Only tested with a powered USB hub</td>
</tr>
<tr>
<td>13fe</td>
<td>5700</td>
<td>Kingston Technology Company Inc USB Stick from Microcenter</td>
<td></td>
<td>DO NOT USE, I/O errors formatting</td>
</tr>
</tbody>
</table>

### 3.3.2. Networking

Raspberry Pi 4B note: For performance reasons, we recommend using the on-board gigabit ethernet port for ESXi host networking.

<table>
<thead>
<tr>
<th>VID</th>
<th>PID</th>
<th>Description</th>
<th>Product</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>0bda</td>
<td>8153</td>
<td>Realtek RTL8153 Gigabit Ethernet Adapter</td>
<td><a href="https://www.amazon.com/gp/product/B01J6583NK">https://www.amazon.com/gp/product/B01J6583NK</a></td>
<td>Not recommended with a stock power supply on the Raspberry Pi 4B</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><a href="https://www.amazon.com/gp/product/B00BBD7NFU">https://www.amazon.com/gp/product/B00BBD7NFU</a></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><a href="https://www.amazon.com/gp/product/B01KA0UR3O">https://www.amazon.com/gp/product/B01KA0UR3O</a></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><a href="https://www.amazon.com/gp/product/B01M7PL2WP">https://www.amazon.com/gp/product/B01M7PL2WP</a></td>
<td></td>
</tr>
</tbody>
</table>

**Note:** USB RTL8153 NIC will show up in ESXi-Arm incorrectly as 100Mbit, although speeds can be faster it can be constrained further when using it with Raspberry Pi due to hardware/software constraints.
3.3.3. Keyboards

Generally, any USB keyboard should work.

<table>
<thead>
<tr>
<th>VID</th>
<th>PID</th>
<th>Description</th>
<th>Product</th>
</tr>
</thead>
</table>

3.4. Supported plug-in PCIe adapters

Every other platform outside of the Raspberry Pi requires a bring-your-own solution for networking. It is recommended that PCIe NICs are used where possible. While most platforms have on-board SATA controllers, plug-in SATA adapters are supported as well. The tested drivers are a subset of the 7.0.

<table>
<thead>
<tr>
<th>Type</th>
<th>Vendor</th>
<th>Driver</th>
<th>Comments</th>
</tr>
</thead>
</table>

4. Preparation

Follow the hardware-specific guides around configuring the system.

4.1. iSCSI

This is a non-exhaustive guide to using iSCSI and installing ESXi to an iSCSI LUN.

4.1.1. Sorting out your LUNs.

Everyone’s SAN is a bit different.

4.1.1.1. QNAP

Note: QNAP’s LUNs start at 0.
The most important item here is the IQN. For the sake of this example, authentication is left disabled.

**Quick Configuration Wizard**

**CHAP Authentication Settings**

- Use CHAP authentication
  - Username:
  - Password:
  - Re-enter Password:

I do set a few advanced settings and these appear to be okay.

**Advanced Settings**

- 4K bytes Sector Size
- Report volatile write cache (recommended for data safety)
- FUA bit capability (recommended for data safety)

**Step 6/11**

4.1.1.2. Synology

Synology tells you what number to use for your created volumes. If you have multiple LUNs in the same target, you need to match the value in **Number** column.

**Note:** Synology LUNs start at 1.

4.1.1.3. Others

FreeNAS, etc, configuration will be different. You may have different or additional settings. Experiment, especially if you start seeing “strange” behavior, such as the target not visible in UEFI, or having “read-only” access in ESXi, such as the installer failing to format the LUN.
4.1.2. Configuring firmware

The following directions apply to any system using Tianocore edk2-based UEFI, provided iSCSI support is compiled into the firmware. For systems with commercial UEFI, the UI will be different.

Start at the main UEFI setup page. On the Pi, you can reach this screen by mashing the ESC key. Use arrow keys to select Device Manager.

Press ENTER. Now select iSCSI Configuration.

Press ENTER. Now select the iSCSI Initiator Name field. Press ENTER. A text entry box will pop up. Enter something here that matches IQN format (usually in the format iqn.year-month.your.domain:machine-id). Press ENTER.
Now select **Add an Attempt** and press **ENTER**.

Select the NIC that you will use for the iSCSI boot attempt. On the Pi, there’s only one on-board NIC, so just press **ENTER**.

Now, navigate to the **iSCSI Mode** field, press **ENTER**, and select **Enabled**, pressing **ENTER** to complete selection. For the sake of this example, use IPv4. Navigate to **Enable DHCP** and press **SPACEBAR** to enable.
This is the most important info: enter the IQN correctly under **Target Name**. If you mistype it here, don’t correct it as it won’t “stick”. Exit out and try re-entering the entire form (it’s a TianoCore bug). You also need the server connection info and the LUN ID.

For the sake of keeping things simple, disable CHAP.

Now navigate to **Save Changes**.
Press **ENTER**. You should see your added iSCSI boot attempt listed.

Exit out (**ESC**) all the way to the main setup screen and select **Reset**.

Press **ENTER**. Your Pi will reboot. Hit **ESC** to enter UEFI setup again. It will take a bit of time (it's connecting to the iSCSI target). Select **Boot Manager**.
Press ENTER. You should see the iSCSI target listed here.

Of course now you can boot the ESXi installer (e.g from USB drive).

4.1.3. What could go wrong?

If there are networking connectivity issues the iSCSI boot option will not be listed in the Boot Manager. This includes obvious stuff (NIC cable fell out, SAN is off) but also invalid IQN configuration.

Note that the attempt is still listed in the iSCSI Configuration portion of UEFI setup!
Important: if you made any errors in the config, delete and re-create the attempt. There’s a UEFI bug where the attempt configuration won’t be updated.

As soon as the problem is fixed (i.e. NIC cable is back), the boot option will re-appear, even at the same spot/ordering as it was before.

Note: if you got the wrong LUN number, the entry may appear but be non-functional.

Also, if you configure iSCSI with DHCP, note that the boot entry will change if the DHCP offer (IP address) changes. This means that the boot order will change, as the old entry will be removed and the new entry added at the bottom of the list. Caution!

5. Create ESXi Installer USB Key

For this you’ll need the ESXi-Arm Fling ISO, of course.

5.1. Using balenaEtcher on Linux, Mac or Windows

Download balenaEtcher from https://www.balena.io/etcher/, and choose the Fling ISO and USB key target. Hit Flash!

You can ignore the warning about the missing partition table, it’s an ISO.
⚠️ Missing partition table

It looks like this is not a bootable image. The image does not appear to contain a partition table, and might not be recognized or bootable by your device.

Cancel  Continue

Select target  2 found

- Name
- SanDisk Ultra Media  63.5 GB  /dev/disk3

Show 1 hidden

Cancel  Select (1)
5.2. On macOS

Identify the disk using the following command and make note of the disk path (e.g., /dev/diskX), and make sure any existing partitions are unmounted.
$ diskutil list
/dev/disk4 (external, physical):
#:                       TYPE NAME                    SIZE       IDENTIFIER
0:      GUID_partition_scheme                        *256.6 GB   disk4
1:                        EFI EFI                     209.7 MB   disk4s1
2:                  Apple_HFS Untitled                256.3 GB   disk4s2
$ diskutil unmount /dev/disk4s2
Volume Untitled on disk4s2 unmounted
$ diskutil unmount /dev/disk4s1
disk4s1 was already unmounted

Raw-write the ISO file to the drive, using the disk identified above. Note the use of the raw device (/dev/rdisk4, not /dev/disk4)

$ sudo dd if=~/.VMware-VMvisor-Installer-7.0.0-16966451.aarch64.iso of=/dev/rdisk4 bs=$((16 * 1024 * 1024))

Eject the drive:
$ hdiutil eject /dev/disk4

6. Installing ESXi-Arm

Make sure to also follow the notes in the the hardware-specific guides for installation and post-installation steps.

- Fling on Raspberry Pi
- Fling on Ampere eMAG 8180-based Servers
- Fling on Arm Neoverse N1 SDP
- Fling on SolidRun HoneyComb LX2K
- Fling on NXP LS1046A FRWY
- Fling on NXP LS1046A RDB

6.1. Basic installation

After booting the installer ISO, you will see the installer:

After accepting the EULA, the installer will list available storage media for installation. Use the arrow keys to select the drive to install it.
Select your keyboard layout:

Choose a password:

Press F11 to confirm the install.

Note: If you're using the Raspberry Pi USB keyboard, F11 is the combination of Fn and F1.
Installation should be complete. Press ENTER to reboot.

6.2. Operation with a video console

If you're installing ESXi-Arm on a system with a video adapter (or you're plugging a screen into a Raspberry Pi), ESXi will default to using the video and USB keyboard for its console.

This how DCUI (console UI) looks on boot-up:
The video console has a notion of virtual terminals, which can be switched between using key combos:

<table>
<thead>
<tr>
<th>Key combo</th>
<th>Description</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALT-F2</td>
<td>DCUI</td>
<td>Or installer when installing</td>
</tr>
<tr>
<td>ALT-F1</td>
<td>ESXi Shell</td>
<td>If ESXi Shell access is enabled</td>
</tr>
<tr>
<td>ALT-F11</td>
<td>Kernel status</td>
<td></td>
</tr>
<tr>
<td>ALT-F12</td>
<td>Kernel log messages</td>
<td>dmesg</td>
</tr>
</tbody>
</table>

6.3. Headless operation

If you’re installing ESXi-Arm on a system without a video adapter (or you’re not plugging a screen into a Raspberry Pi), ESXi-Arm will use a serial port for its console. Usually, systems will have only one such port. In case there are several, system UEFI may have additional Serial Port Console Redirection configuration.

Like a system with a video console, ESXi will boot up to a DCUI (console UI) screen, although this will look a bit different:

ESXi actually supports several different "roles" for the serial port. These roles are like virtual terminals and can be switched between using key combos:

<table>
<thead>
<tr>
<th>Key combo</th>
<th>Description</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTRL-G CTRL-B 3</td>
<td>DCUI</td>
<td>Or installer when installing</td>
</tr>
<tr>
<td>CTRL-G CTRL-B 2 ENTER</td>
<td>ESXi Shell</td>
<td>If ESXi Shell access is enabled</td>
</tr>
</tbody>
</table>
6.4. After installation

Now you're ready to check out the VMware ESXi Host Client.

There are a few settings you can change directly from DCUI (console UI).

To customize configuration press F2. You will be asked to authenticate.

6.4.1. Changing NIC and TCP/IP configuration

*Note:* The system will default to DHCP on the first NIC detected during install that had a link up.

Use arrow keys to navigate to Configure Management Network:

![Authentication Required](image)

Press ENTER. The rest of this should be fairly self-explanatory.

6.4.2. Enabling SSH or console shell

*Note:* ESXi Shell (from console) and SSH access are disabled by default.

Use arrow keys to navigate to Troubleshooting Options:

Press ENTER.
You can toggle ESXi Shell or SSH support by selecting the entry and pressing ENTER.

7. VMware ESXi Host Client

This is a web UI to your installed ESXi-Arm system. This is where you can configure ESXi-Arm and create and access VMs.

7.1. Login and host configuration

2. Setting up NTP: It is especially important to make sure that the host has clock synchronized, if you plan to connect to vCenter later.
   a. Under Host Manage System, select Time & date and click Edit NTP Settings.
   b. Select Use Network Time Protocol (Enable NTP client).
   c. Enter the IP address or fully qualified domain name of one or more NTP servers to synchronize with.
      (You can use pool.ntp.org)
   d. Save
   e. Under Networking Firewall rules, select the NTP Client line, and choose Enable in the Actions menu.
   f. Under Host Manage Services, select the ntpd line and click on Start.

7.2. Virtual Machine Creation

1. Right click Host Select Create/Register VM
2. Select creation type Create a new virtual machine
3. Enter Name and Guest OS details
   a. Compatibility: ESXi 7.0 virtual machine
   b. Guest OS family: Linux
   c. Guest OS version: <any from the below supported guest list>

4. Select storage Standard: <choose from the available datastores> (Note: If you’re using the Pi and installing the ESXi-Arm bits on a usb stick, and the usb stick is <128GB, and you do not see an available datastore at this step, it's possible that you may have neglected the autoPartitionOSDataSize during the initial boot-install. Unfortunately the only way to correct this is to redo the installation. This time make sure to append the field along with the size. See the Pi installation guide Section 4 for details)

5. Customize settings
   a. CPU: <choose from available list>
   b. Memory: <within available limit>
   c. Hard disk: <within available limit>
   d. USB controller: <default> (USB 3.1)
   e. Network Adapter: <default> (E1000e)
   f. CD/DVD Drive: Choose “Datastore ISO file” Browse datastore to upload/find the required ISO
   g. Video Card: <default>

Additional hardware can be added with “Add hard disk”, “Add network adapter” and “Add other device” options.

Note: The USB controller is required to use the keyboard and mouse to interact with the Virtual Machine.
6. Review settings Finish

7. Powering on the VM should take you to the OS installer.

### 7.3. Guest OS support

These operating systems have been tested to properly install with the following basic configuration:

<table>
<thead>
<tr>
<th>Virtual Hardware</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
</tr>
<tr>
<td>Memory</td>
</tr>
<tr>
<td>Hard disk</td>
</tr>
<tr>
<td>SATA Controller</td>
</tr>
<tr>
<td>USB controller</td>
</tr>
<tr>
<td>Network adapter</td>
</tr>
<tr>
<td>CD/DVD Drive</td>
</tr>
<tr>
<td>Video Card</td>
</tr>
</tbody>
</table>

These operating systems support both the UEFI firmware in the virtual machine and the DT (device tree) method of describing virtual machine hardware.

<table>
<thead>
<tr>
<th>Distribution</th>
<th>URL</th>
<th>Verified By</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ubuntu 20.04 LTS</td>
<td><a href="https://ubuntu.com/download/server/arm">https://ubuntu.com/download/server/arm</a></td>
<td>VMware</td>
</tr>
<tr>
<td>CentOS Linux 8</td>
<td><a href="http://isoredirect.centos.org/centos/8/isos/aarch64/">http://isoredirect.centos.org/centos/8/isos/aarch64/</a></td>
<td>VMware</td>
</tr>
<tr>
<td>openSUSE Leap 15.2</td>
<td><a href="http://download.opensuse.org/ports/aarch64/distribution/leap/15.2/isos/">http://download.opensuse.org/ports/aarch64/distribution/leap/15.2/isos/</a></td>
<td>VMware</td>
</tr>
<tr>
<td>Photon OS 3.0</td>
<td><a href="https://github.com/vmware/photont/wiki/Downloading-Photon-OS">https://github.com/vmware/photont/wiki/Downloading-Photon-OS</a></td>
<td>VMware</td>
</tr>
<tr>
<td>Debian 10.x</td>
<td><a href="https://cdimage.debian.org/debian-cd/current/arm64/iso-cd/">https://cdimage.debian.org/debian-cd/current/arm64/iso-cd/</a></td>
<td>VMware</td>
</tr>
<tr>
<td>FreeBSD</td>
<td><a href="https://vincerants.com/freebsd-under-vmware-esxi-on-arm-fling/">https://vincerants.com/freebsd-under-vmware-esxi-on-arm-fling/</a></td>
<td>Community</td>
</tr>
<tr>
<td>NetBSD</td>
<td><a href="https://twitter.com/jmcwhatever/status/1317138758559465472">https://twitter.com/jmcwhatever/status/1317138758559465472</a></td>
<td>Community</td>
</tr>
<tr>
<td>Alpine</td>
<td><a href="https://alpinelinux.org/downloads/">https://alpinelinux.org/downloads/</a></td>
<td>Community</td>
</tr>
<tr>
<td>Devuan</td>
<td><a href="https://devuan.org/get-devuan">https://devuan.org/get-devuan</a></td>
<td>Community</td>
</tr>
</tbody>
</table>
7.4. See also
- Add a CD or DVD Drive to a Virtual Machine in the VMware Host Client: https://docs.vmware.com/en/VMware-vSphere/7.0/com.vmware.vsphere.hostclient.doc/GUID-132901CE-AE24-4A61-B8CB-55CF6489A8A7.html
- General guide to managing a VM from VMware Host Client: https://docs.vmware.com/en/VMware-vSphere/7.0/com.vmware.vsphere.hostclient.doc/GUID-4ECD8CE7-6362-4FC3-A2DA-CD3D6882306.html

8. VMware vCenter Server Basics

The vCenter Server and Host Management full documentation is available at https://docs.vmware.com/en/VMware-vSphere/7.0/com.vmware.vsphere.vcenterhost.doc/GUID-3B5AF2B1-C534-4426-B97A-D14019A8010F.html

8.1. Add a Datacenter and a Cluster

When vCenter Server is just installed you need to create a Datacenter object, and eventually a Cluster to add your ESXi hosts to.

8.1.1. Adding a Datacenter

On the Host and Clusters page, select the VC address and click on New Datacenter... in the ACTIONS drop down menu. Choose a name and validate the popup.

8.1.2. Adding a Cluster

Still on the Host and clusters page, select the Datacenter and click on New cluster... in the ACTIONS drop down menu. Choose a name and validate the popup.
For more information on clusters, please see the *Creating and Configuring Clusters* documentation: https://docs.vmware.com/en/VMware-vSphere/7.0/com.vmware.vsphere.vcenterhost.doc/GUID-F7818000-26E3-4E2A-93D2-FCDCE7114508.html

8.2. 2. Add an ESXi-Arm Fling host to vCenter Server

On the *Hosts and clusters* page, select the Datacenter and click on **Add Hosts...** in the **ACTIONS** drop down menu.

1. Name and location: enter the IP address of the host
2. Connection settings: enter the host credentials
3. Host summary: review the information
4. Assign license: choose a license to use for the host
5. Lockdown mode: let on Disabled (default)
6. VM location: choose the Datacenter
7. Ready to complete: Click Finish to add the host

Here is an Ampere Computing eMAG added to the Datacenter:

8.3. 3. Create a Virtual Machine
On a Datacenter, a Cluster or a Host, choose New Virtual Machine... from the ACTIONS drop down menu.

1. Select creation type: Create a new virtual machine
2. Enter Name and a location for the virtual machine
3. Select a compute resource: the host
4. Select storage: datastore1
5. Select compatibility: ESXi 7.0 and later
6. Select a guest OS:
   a. Guest OS Family: Linux
   b. Guest OS Version: <any from the below supported guest list>
7. Customize hardware:
   a. CPU: <choose from available list>
   b. Memory: <within available limit>
   c. Hard disk: <within available limit>
   d. Network Adapter: <default> (E1000e)
   e. USB controller: <default> (USB 3.1)
   f. CD/DVD Drive: Choose *Datastore ISO file* Browse datastore to upload/find the required ISO and click on Connect At Power On.
   g. Video Card: <default>
   **Note:** The USB controller is required to use the keyboard and mouse to interact with the Virtual Machine.
8. Ready to complete Finish

Powering on the VM should take you to the OS installer:
8.4. 4. Access VM Remote Console

Clicking on Launch Web Console opens a new browser window with the Web Console. Use the keyboard and mouse to interact with the VM.
9. Enabling vMotion

9.1. Tested Platforms

<table>
<thead>
<tr>
<th>Platform</th>
<th>Supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ampere eMAG</td>
<td>Yes</td>
</tr>
<tr>
<td>HoneyComb LX2K</td>
<td>Yes</td>
</tr>
<tr>
<td>NXP FRWY</td>
<td>Yes</td>
</tr>
<tr>
<td>Raspberry Pi</td>
<td>Yes</td>
</tr>
</tbody>
</table>

General guide to vMotion on virtual machines:

Note: Do not mix systems in the same cluster. E.g. do not mix eMAGs and Pies in the same cluster. Also, do not mix x86 and Arm systems in the same cluster.

9.2. Pre-requisites

1. It is recommended that a separate NIC be configured for vMotion and FT logging to ensure that sufficient bandwidth is available.
   a. From the vSphere Web Client navigate to the host Configure Networking Vmkernel adapters
   b. Choose the vmkernel port group to be configured for vMotion and select Edit
   c. Enable vMotion from the list of available services

2. When you migrate virtual machines with vMotion and choose to change only the compute host, the VM needs to be on shared storage to ensure that it is accessible to both source and target hosts. Shared storage can be configured with a SAN, or implemented using iSCSI and NAS.

9.3. VM migration with vMotion

1. Navigate to the VM you want to migrate, right-click and select Migrate...
2. Select a migration type
   a. Change compute resource only (VM needs to be on shared storage)
   b. Change storage only (Migrate VM to a different datastore, but same host)
c. Both (Migrate VM to a different host and datastore)

3. Select destination host

4. Select destination storage
5. Select destination network

6. Select vMotion Priority: <recommended>

7. Review and finish

10. Enabling vSphere HA

10.1. Tested Platforms
1. Platform Supported
   - Ampere eMAG: Yes
   - HoneyComb LX2K: Yes
   - NXP FRWY: Yes
   - Raspberry Pi 4B: Yes

10.2. Pre-requisites

   1. Download the specific FDM VIB from the ESXi-Arm Fling site for your version of your vCenter Server. At launch of the fling, vCenter Server 7.0d (Build 16749653) and vCenter Server 7.0c (Build 16620007) are supported.

   **Step 1** - Upload the FDM VIB to ESXi host via SCP or vSphere Datastore Browser

   **Step 2** - Install the FDM VIB

   ```
   [root@rpi-2.primp-industries.com:~] esxcli software vib install -v /vmware-fdm-7.0c-16620014.arm64.vib --no-sign-check
   Installation Result
   Message: Operation finished successfully.
   Reboot Required: false
   VIBs Installed: VMware_bootbank_vmware-fdm_7.0.0-1620014
   VIBs Removed:
   VIBs Skipped:
   ``

   **Step 3** - Enable vSphere HA on the vSphere Cluster

   ![Edit Cluster Settings](image)

   **Step 4** - If you wish to get rid of "The number of vSphere HA heartbeat datastores for this host is 1, which is less than required: 2" message, you can add the following Advanced Setting `das.ignoreInsufficientHbDatastore = true` and then right click on one of the ESXi hosts and select "Reconfigure for vSphere HA" operation the message to go away.
11. Enabling vSphere Fault Tolerance

11.1. Tested platforms

<table>
<thead>
<tr>
<th>Platform</th>
<th>Supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ampere eMAG</td>
<td>Yes</td>
</tr>
<tr>
<td>HoneyComb LX2K</td>
<td>Untested, should work</td>
</tr>
<tr>
<td>NXP FRWY</td>
<td>No</td>
</tr>
<tr>
<td>Raspberry Pi 4B 8GB</td>
<td>Yes</td>
</tr>
<tr>
<td>Raspberry Pi 4B 4GB</td>
<td>No</td>
</tr>
</tbody>
</table>

11.2. Pre-requisites

0. Suggested reading - General guide to FT on virtual machines: https://docs.vmware.com/en/VMware-vSphere/7.0/com.vmware.vsphere.avail.doc/GUID-7525F8DD-9B8F-4089-B020-BAA4AC6509D2.html

1. It is recommended that a separate NIC be configured for vMotion and FT logging to ensure that sufficient bandwidth is available.
   a. From the vSphere Web Client navigate to the host Configure Networking Vmkernel adapters
   b. Choose the vmkernel port group to be configured for FT and select Edit
   c. Enable vMotion and Fault Tolerance logging from the list of available services (Yes, you need both)

2. vSphere HA should be enabled for the cluster. See Enabling vSphere HA section for detailed instructions

Step 1 - Browse to the VM in vSphere web client Right-click and select Fault Tolerance Turn On Fault Tolerance
Step 2 - Select a datastore where the Secondary VM is to be placed

Step 3 - Select a host where the Secondary VM is to be placed

Step 4 - Review and finish

Step 5 - Testing FT with failover

1. Make sure the VM configured for FT is turned on
2. Browse to the VM in vSphere web client Right-click and select Fault Tolerance Test Failover
   (The secondary VM in the second host should now be the primary VM)
12. VMware Tools

ESXi-Arm does not include an installable version of VMware Tools for Arm Guest Operating Systems. To install VMware Tools, you will need to manually compile Open VM Tools for your specific Guest Operating System.

Here is an example of compiling latest Open VM Tools 11.1.5 for Ubuntu 20.04 AARCH64. For VMware Photon OS AARCH64, you can refer to this blog post for instructions.

**Step 1** - Update and Install the following package dependencies:

```bash
apt update
apt install -y automake-1.15 pkg-config libtool libmspack-dev libglib2.0-dev
   libpam0g-dev libssl-dev libxml2-dev libxmlsec1-dev libx11-dev libxext-dev
   libxinerama-dev libxi-dev libxrender-dev libxrandr-dev libgdk2.0-dev
   libgdk-3-dev libgtkmm-3.0-dev
```

**Step 2** - Clone Open VM Tools git repo:

```bash
git clone https://github.com/vmware/open-vm-tools.git
cd open-vm-tools/open-vm-tools/
```

**Step 3** - Run the following commands to build and install Open VM Tools:

```bash
autoreconf -i
./configure
sudo make
sudo make install
```

**Step 4** - We need to create a systemd unit file so we can enable and start Open VM Tools Daemon upon startup. Run the following command to create `vmtoolsd.service` file:

```bash
cat > /etc/systemd/system/vmtoolsd.service << EOF
[Unit]
Description=Open VM Tools
After=network-online.target

[Service]
ExecStart=/usr/local/bin/vmtoolsd
Restart=always
RestartSec=1sec

[Install]
WantedBy=multi-user.target
EOF
```

**Step 5** - Enable and Start Open VM Tools Daemon and verify using either the ESXi Host Client UI or vSphere UI that Open VM Tools is now running:

```bash
systemctl enable vmtoolsd.service
systemctl start vmtoolsd.service
```
13. Applications and Functionality to Test on ESXi-Arm

13.1. Datacenter and Near Edge

For larger capacity servers such as those based on the Ampere Computing eMAG 8180, a number of system benchmarking tools including https://github.com/ARM-software/meabo exist to help you understand the capabilities and performance you can obtain on your system. VMware also offers Weathervane that can be used to stress test K8 clusters.

13.2. Mid-Edge (HoneyComb LX2K)

For smaller mid-range hardware, you can test one of several Edge computing Cloud framework such as Amazon's AWS Greengrass platform which is an IoT Edge processing offering integrated with AWS Cloud services and capable of running offline or when networking is intermittent. Greengrass allows execution of Lambdas, ML models, event processing and more. You can find out more about the supported Arm platforms here. Similarly, Azure IoT Edge is the Microsoft Azure Edge processing platform for IoT offering a range of services such as Azure functions, streaming analytics, ML models and more. Azure IoT Edge supports multiple Arm platforms including Arm64. You can find out more details on the supported Arm platforms here.

13.3. Far Edge (Raspberry Pi 4, NXP FRWY)

For the RPi4, check out the various Arm compatible OS options available including Photon OS and we suggest testing various VM lifecycle operations such as power events, snapshots, maintenance mode, vMotion, creation of clusters. Exercise the use of resource reservations to ensure the limited CPU & memory resources go to the highest priority VMs. Test VMware HA by simulating a node failure if you have a cluster, and even test out Fault Tolerance for a zero-downtime far Edge solution. We hope these suggestions are helpful but we know you’ll have many of your own ideas about how to use this technology and we look forward to hearing about your use-cases and experience.

14. Troubleshooting

14.1. Support

If you are running into installation or setup issues with the ESXi-Arm Fling, please use the Comments and/or Bug section of the ESXi-Arm Fling website. In addition, you can also engage with the ESXi-Arm team and community on Slack at #esxi-arm-fling on VMware (code)

14.2. Generating Support Bundle

If you are able to install ESXi-Arm Fling and still having issues, it is recommended that you provide a support bundle which can be accessed by Engineering. You can either share a public download URL in the Comments/Bugs section of the ESXi-Arm Fling website or you can directly share that with someone from ESXi-Arm team via private Slack direct message.

There are two methods to generate support bundle, using either the ESXi Host Client UI or ESXi Shell.

14.2.1. ESXi Host Client UI
Open browser and login to the ESXi-Arm IP Address/Hostname. Under Actions, select "Generate support bundle" and once the support bundle has completed, you will be provided a download link.

14.2.2. ESXi Shell

SSH to ESXi-Arm IP Address/Hostname and then type vm-support. Once the support bundle has completed, you will need to SCP it off the ESXi-Arm host to your local desktop.

15. Known issues

15.1. ESXi

15.1.1. Network and disk monitoring may not be accurate

This is largely dependent on I/O drivers, some of which are in-development.

15.1.2. Virtualization

15.1.2.1. Performance Monitoring Counter support in VM

PMUv3 is not implemented, and is not advertised to the guest. Linux does not attempt to use it, however if another Guest OS tries to access it, the Virtual Machine will stop immediately.

15.1.2.2. vMotion is not supported between hardware with different SoCs

Do not attempt to migrate a running Virtual Machine between different systems (e.g. Ampere eMAG and a Raspberry Pi).

15.1.2.3. ACPI operating systems are not supported

The virtual machine only models DT (device tree) today.

15.1.2.4. Fixed-size frame buffer graphics

The UEFI GOP frame buffer defaults to 1024x768. It is also not accelerated.

15.1.2.5. Guest VM hangs with black screen after a while

A Linux VM may be configured to enter a low power mode (aka "suspend to idle", nothing to do with suspending VMs).
**Workaround:** Make sure automatic suspend is disabled in your desktop settings.

15.1.2.6. **Paravirtualized drivers are not built in Linux**

VMXNET3, VMCI, PVSCSI, SVGA3 and balloon drivers are generally not available in standard Linux distributions. With the exception of Ubuntu having VMXNET3 support.

15.1.2.7. **USB1.1 is not supported in Debian**

Debian does not come with the ohci-hcd driver, and will not detect the keyboard and pointing devices if virtual USB1.1/USB2 controllers are enabled.

**Workaround:** Only use virtual USB3 controller (which is the default when creating a Virtual Machine).

15.2. **vCenter**

15.2.1. **A general system error occurred: Unable to push signed certificate to host**

The warning message is shown in vSphere UI when adding ESXi-Arm host to vCenter Server. This occurs as there is a time skew between the ESXi-Arm host and vCenter Server, and is exacerbated due to some systems (e.g. Raspberry Pi) not having a battery backed RTC.

**Workaround:** Ensure all systems sync their time from the same source. For detailed instructions on configuring NTP for ESXi-Arm host, please refer to the “VMware ESXi Host Client” section.

15.2.2. **vSphere Auto Deploy is not supported for arm64 hosts**

The vSphere Auto Deploy can not be used with arm64 hosts.