Running Graphical Programs on Windows Subsystem on Linux

Pushing the boundaries of what WSL can do.

By Tom Fenton   **02/08/2017**

This is the third article in a series on using Microsoft Windows Subsystem on Linux (WSL). The **first article (https://virtualizationreview.com/articles/2017/01/12/installing-ubuntu-on-windows-10-on-vsphere.aspx)** discussed how to enable Linux on Windows and install it on an ESXi server. The **second article (https://virtualizationreview.com/articles/2017/01/27/using-the-windows-subsystem-for-linux.aspx)** covered some of the things you can do with WSL after it's been installed. This final chapter will cover using graphical programs on it.

Running OpenSSH Server

WSL comes with the OpenSSH package, but because it's only a minimal package, it will need to be uninstalled, and the full OpenSSH server installed and configured **(Figure 1).**

Fortunately, WSL comes with apt-get, making it a trivial process to install the full version:

```
#sudo apt-get remove openssh-server

#sudo apt-get install openssh-server
```


(~/media/ECG/virtualizationreview/Images/2017/02/0117vrm_Fenton3Fig1.ashx)
[Click on image for larger view.]
Figure 1. Installing OpenSSH Server.

Windows 10 comes with an SSH server, and it will either need to be disabled or the port that will be used for SSH on WSL will need to be changed. I chose to do the latter and use port 2200 for WSL SSH.

A few other parameters will also need to be changed in the OpenSSH configuration file. The file -- /etc/ssh/sshd_config -- will need to be edited to add/change the following:

Change - PermitRootLogin no
Add - AllowUsers yourusername
Change - PasswordAuthentication yes
Add - UsePrivilegeSeparation no
Change - ListenAddress 0.0.0.0
Change - Port 2200

After these changes have been made, the SSH service will need to be restarted:

```
# sudo service ssh --full-restart
```
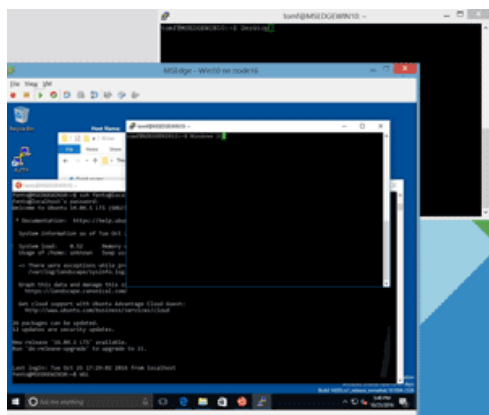
Any firewalls running on the Windows side will need to be disabled or have a port opened up.

Figure 2 is a screenshot of connecting to the WSL SSH server directly from WSL from the Windows 10 desktop using **PuTTY (http://www.putty.org/)**, and my desktop using PuTTY.

(**~/media/ECG/virtualizationreview/Images/2017/02/0117vrm_Fenton3Fig2.ashx**)

[Click on image for larger view.]
Figure 2. Connecting to WSL SSH server.

Note: If you're having difficulty getting OpenSSH Server running, you can always try another SSH server such as Dropbear. To install and use dropbear do the following:

```
#sudo apt-get install dropbear

#sudo dropbear
```

Run Graphical Programs

Microsoft doesn't support graphical programs on WSL. Bash on WSL is intended for running command-line programs that developers might need, but it's possible to run graphical Linux desktop programs on Windows using the Bash shell. To be more precise, you'll be able to display graphical programs running in WSL on a Windows 10 desktop by using an X server which runs on Windows 10.

To use WSL with graphical programs, an X server will need to be installed on the Windows 10 system and the DISPLAY variable will need to be set in Bash. Fortunately, there are many X servers that run on Windows; some of the more popular free ones are **Xming (../../../../Users/Dave/Downloads/vdc-us04-a.vmeduc.com)**, **Cygwin X (http://x.cygwin.com/)**, and **vcXsrv (https://sourceforge.net/projects/vcxsrv/)**.

I've used **Xming (https://sourceforge.net/projects/xming/)** and have had good luck with it in the past. It comes with a setup wizard, and I installed it on my Windows 10 system without any difficulty (**Figure 3**).

(**~/media/ECG/virtualizationreview/Images/2017/02/0117vrm_Fenton3Fig3.ashx**)

[Click on image for larger view.]
Figure 3. Installing Xming.

I used the default settings; after launching it, Xming appeared in my system tray, running in the background and waiting for a graphical WSL program (**Figure 4**).

(~/media/ECG/virtualizationreview/Images/2017/02/0117vrm_Fenton3Fig4.ashx)
[Click on image for larger view.]
*Figure 4. Xming in the system tray.*

Trying Out Graphics Applications

Once you have an X server installed and running, you'll need to install graphics applications. A good starting point would be the x11-apps package, which has about two dozen X applications, including xclock, xcalc, and xeyes. To download and install it run the following:

```
# sudo apt-get install x11-apps
```

Once the applications have been installed, you can start them by setting your display and executing the application on the Bash shell. **Figure 5** shows xeyes running (a program with "eyes" that follow a moving cursor).

```
# export DISPLAY=:0

# xeyes
```
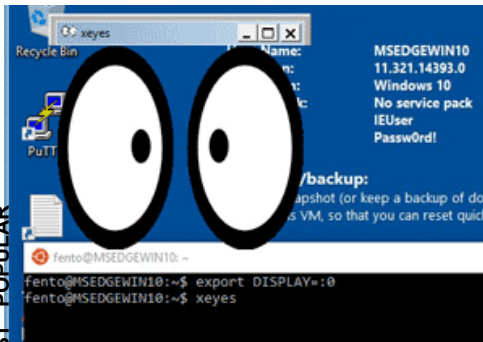

(~/media/ECG/virtualizationreview/Images/2017/02/0117vrm_Fenton3Fig5.ashx)
[Click on image for larger view.]
*Figure 5. Running an **X** application.*

Three other applications that I worked with were Blockout (a 3-D Tetris-like game), VLC (a video player), and the Firefox Web browser. All of them were installed using apt-get:

```
# sudo apt-get install blockout2

# sudo apt-get install vlc

# sudo apt-get install firefox
```

Firefox (**Figure 6**) worked even though I received some error messages on the command line when I first started it. Blockout (**Figure 7**) worked flawlessly, which surprised me as it is an extremely graphics-intensive application.
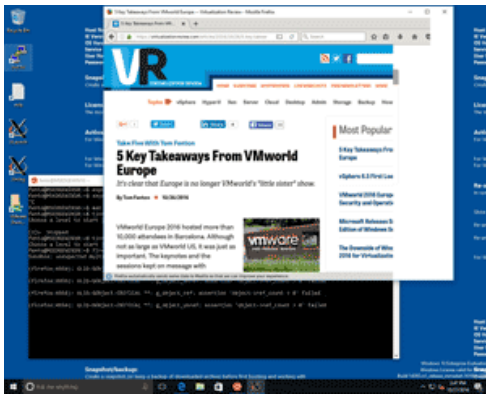
(~/media/ECG/virtualizationreview/Images/2017/02/0117vrm_Fenton3Fig6.ashx)

[Click on image for larger view.]

Figure 6. Running Firefox.
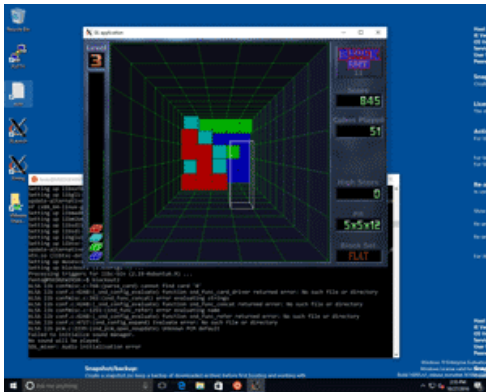
(~/media/ECG/virtualizationreview/Images/2017/02/0117vrm_Fenton3Fig7.ashx)

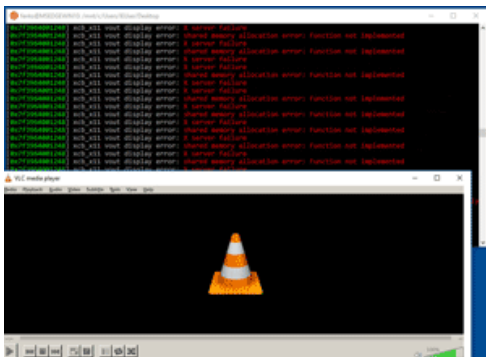**[Click on image for larger view.]**

Figure 7. Running Blockout2.

When I tried to run VLC, many error messages were generated when I tried to load an MP4 file. The VCL interface and error messages are shown in **Figure 8**.

Microsoft has a very good Web server and doesn't recommend running one on WSL, but I was able to install, configure, and run the Apache2 server, albeit with some warning on startup, and while it was running.

```
# sudo apt-get install apache2


# sudo /etc/init.d/apache2 start
```

I did try other graphic applications such as twm, xterm, tightvncserver and xman, and they worked fine, while other applications such as xprop, xmag, xload and xconsole didn't.

(~/media/ECG/virtualizationreview/Images/2017/02/0117vrm_Fenton3Fig8.ashx)

[Click on image for larger view.]

Figure 8. Trying (unsuccessfully) to run VLC.

Add WSL to Your Toolbox

I honestly didn't expect the X graphics system to work on WSL, and although it's far from being ready for production, I was

able to run some fairly sophisticated and powerful programs using it.

In this three-part series I first walked you through downloading and installing Windows 10 on vSphere. I then showed you how to use it for what it was designed for, and finally I pushed its limits to do things it was expressly designed *not* to do. This demonstrates that WSL is a valuable tool that you may want to install and become familiar with. Although the graphics programs are inconsistent, the rich set of text line tools make it a valuable tool for developers and casual users alike.

About the Author

Tom Fenton works in VMware's Education department as a Senior Course Developer. He has a wealth of hands-on IT experience gained over the past 20 years in a variety of technologies, with the past 10 years focused on virtualization and storage. Before re-joining VMware, Tom was a Senior Validation Engineer with The Taneja Group, were he headed their Validation Service Lab and was instrumental in starting up its vSphere Virtual Volumes practice. He's on Twitter @vDoppler.

PRINTABLE FORMAT (HTTPS://VIRTUALIZATIONREVIEW.COM/ARTICLES/2017/02/08/GRAPHICAL-PROGRAMS-ON-WINDOWS-SUBSYSTEM-ON-LINUX.ASPX?P=1)

Recommended: Solving VMware Backup Challenges with the Cloud (https://virtualizationreview.com/webcasts/2018/07/druva-jul25.aspx? partnerref=link)
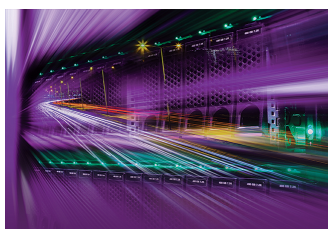
Featured

(https://virtualizationreview.com/articles/2018/12/30/5-eye-catching-technologies-at-kubecon-north-america-2018.aspx)
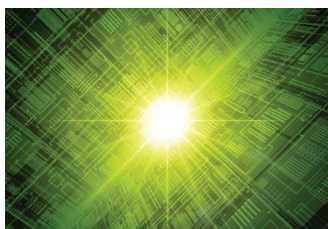
5 Most Eye-Catching Technologies at KubeCon + CloudNativeCon North America 2018 (https://virtualizationreview.com/articles/2018/12/30/5-eye-catching-technologies-at-kubecon-north-america-2018.aspx)

(https://virtualizationreview.com/articles/2018/12/18/aws-pinpoint-part-2.aspx)

Create a Mailing List with AWS Pinpoint, Part 2: Set up a Segment and Launch a Campaign (https://virtualizationreview.com/articles/2018/12/18/aws-pinpoint-part-2.aspx)

(https://virtualizationreview.com/articles/2018/12/12/azure-ultra-ssd.aspx)

A First Look at Azure Ultra SSD (https://virtualizationreview.com/articles/2018/12/12/azure-ultra-ssd.aspx)

CONTACT US (HTTPS://VIRTUALIZATIONREVIEW.COM/PAGES/CONTACT-US.ASPX)

ADVERTISE (HTTPS://1105ECG.COM/PAGES/VR.ASPX)

ARCHIVES (HTTPS://VIRTUALIZATIONREVIEW.COM/ARTICLES/LIST/ARCHIVES.ASPX)

EVENTS (HTTP://TECHMENTOREVENTS.COM/)

PRINT ISSUES ONLINE (HTTPS://VIRTUALIZATIONREVIEW.COM/ISSUES/ISSUE-ARCHIVE.ASPX)

FREE NEWSLETTERS (HTTPS://NEWSLETTERS.1105PUBS.COM/NL/URMG.DO?PKWSNVZR)

(https://converge360.com/Home.aspx)

(https://1105media.com/Home.aspx)

Application Development Trends (https://adtmag.com/home.aspx)

AWSInsider.net (https://awsinsider.net/home.aspx)

Enterprise Systems (https://esj.com/home.aspx)

IoTDev360 (https://iotdev360.com/home.aspx)

Live! 360 (https://live360events.com/home.aspx)

MCPmag.com (https://mcpmag.com/home.aspx)

Prophyts (https://prophyts.com/home.aspx)

Pure AI (https://pureai.com/home.aspx)

Redmond (https://redmondmag.com/home.aspx)

Redmond Channel Partner (https://rcpmag.com/home.aspx)

TechMentor Events (https://techmentorevents.com/home.aspx)

Visual Studio Magazine (https://visualstudiomagazine.com/home.aspx)

Visual Studio Live! (https://vslive.com/home.aspx)

MOST POPULAR