



Reactor

Microsoft Reactor
Get Started with Windows Subsystem for Linux

Event Code: #13071



FAQ's

What are the Reactors?

Reactors are community spaces where technology professionals meet, learn, and connect—to both their local peers as well as industry-leading ideas and technology from Microsoft, partners, and the open source community.

With a diverse mix of hands-on workshops, expert panels, and community events, there's something for everyone—whether you're just getting started or working on complex projects.

Reactor programming is always free and inclusive of a broad set of products, tools, and technologies.

You will find:

- First party content designed to help developers learn new skills in high-demand fields such as Machine Learning, AI, and Data Science
- New content around breakthrough technologies and concepts such as quantum computing and blockchain
- Resources, talks and classes from Microsoft for Startups to deliver what startups need to succeed
- Cloud Advocates who deep-dive into their technical specialties and the Industry Experiences team to provide tailored content around applications for Manufacturing, Healthcare, Retail, and other verticals
- Reactors always welcome community groups, Meetups, partners and MVPs to use our space

Map



Our Code of Conduct

Microsoft is dedicated to empowering every person and every organization on the planet to achieve more.

This includes Microsoft Reactor events where we seek to provide a respectful, friendly, professional experience for everyone, regardless of gender, sexual orientation, physical appearance, disability, age, race or religion.

We do not tolerate any behaviour that is harassing or degrading to any individual, in any form. Individuals are responsible for knowing and abiding by these standards. We encourage everyone to assist in creating a welcoming and safe environment.



Be aware of others



Be friendly and patient



Be welcoming and respectful



Be open to all questions and viewpoints



Be understanding of differences



Be kind and considerate to others

Our Code of Conduct

Microsoft is dedicated to empowering every person and every organization on the planet to achieve more.

This includes Microsoft Reactor events where we seek to provide a respectful, friendly, professional experience for everyone, regardless of gender, sexual orientation, physical appearance, disability, age, race or religion.

We do not tolerate any behaviour that is harassing or degrading to any individual, in any form. Individuals are responsible for knowing and abiding by these standards. We encourage everyone to assist in creating a welcoming and safe environment.

Our Ask of You



Be aware of others



Be friendly and patient



Be welcoming and respectful



Be open to all questions and viewpoints



Be understanding of differences



Be kind and considerate to others

Get Started with Windows Subsystem for Linux



Allen Sanders

Principal Cloud Architect

Speaker Bio: Allen Sanders is a Principal Cloud Architect with over 25 years of experience in software engineering, architecture and design delivering technology strategy and business solutions in multiple verticals. As a Microsoft Certified Professional, he has taught, led and mentored teams of varying sizes in multiple application transformation efforts. He has a passion for learning – both his own and for others.

Agenda

1	Introduction	6	Knowledge Check
2	Installing a Linux Distribution	7	Summary
3	Windows & Linux File System Integration	8	
4	Adding Node.js	9	
5	WSL & Visual Studio Code	10	

Introduction



What is WSL?

- Feature that can be enabled on modern versions of Windows
- With WSL 2, able to run full Linux natively using Hyper-V
- Provides a GNU/Linux terminal for launching Linux commands

What is WSL?

- Enables navigation between the Windows & Linux file systems
- Supports use of modern dev tools (e.g., Python, Node.js, .NET Core)
- Provides powerful capabilities of both platforms on single machine



Installing a Linux Distribution

<https://aka.ms/WindowsSSLinux-8>



Installing WSL

- Use PowerShell to enable this optional feature (requires restart)
- Once enabled, download and install distribution from Microsoft Store
- Multiple Linux distributions supported

Installing WSL

- On first run of distribution, you'll create an account name & password
- Also, good idea to update the distribution (on start and ongoing)
- In addition, supports installation of custom distributions



Windows & Linux File System Integration



Crossing Over

- Continue to have access to Windows commands in command prompt
- Now also have access to Linux commands through WSL
- Windows binaries can be called from Linux and vice versa using WSL

Crossing Over

- Within the Linux shell, can access Windows file system using `/mnt/c`
- From Windows explorer, can access Linux file system using `\\wsl$` share
- Can optionally map network drive from Windows to Linux file system



Adding Node.js

Installing Node.js et al

- Use cURL to pull latest version of Node.js & npm
- Use apt-get install to install
- Use npm to install Express for HTTP request handling
- node --version, npm --version, and express --version used to confirm



Multiple Node.js Modules Available

Tool	Used for
gm, sharp	Image manipulation, including editing, resizing, compression, and so on, directly in your JavaScript code
PDFKit	PDF generation
validator.js	String validation
imagemin, UglifyJS2	Minification
spritesmith	Sprite sheet generation
winston	Logging
commander.js	Creating command-line applications

Using Node.js

- Can be utilized from command-line (interactively)
- Can be utilized with other tools to stand up full, runnable web app
- Use `express <app name> --view=<view engine>` to scaffold



WSL & Visual Studio Code

Installing VS Code et al

- Use <https://code.visualstudio.com/> to install (or search for download)
- If not already installed, use <https://git-scm.com/download/win> to install Git
- Can also use *sudo apt-get install git* to install git in the Linux shell





Install WSL Extension

- In VS Code, search for and install the “Remote – WSL” extension
- Enables VS Code editor in context of WSL
- Provides support for optimal developer experience
- VS Code allows setting of default terminal to “WSL Bash”

Debugging in VS Code

- Create a debugger config file at launch.json
- Debug | Configure feature will create in a .vscode folder in root
- Set breakpoints & step through code using "Launch Program" under Debug



Knowledge Check

Summary

What Did We Cover?

- Enabling WSL and installing Linux distribution
- Windows & Linux file system integration
- Creating a website with Node.js in WSL
- Setting up VS Code with integration to WSL
- Debugging in VS Code





Next Steps

- <https://aka.ms/WindowsSSLinux-8>
- <https://docs.microsoft.com/en-us/windows/wsl/install-win10>
- <https://daily.dev/blog/5-node-js-express-tutorials-to-go-from-beginner-to-expert>



Reactor

Thank You!

Q&A



<https://www.meetup.com/pro/microsoft-reactor>



[@MSFTReactor](https://twitter.com/MSFTReactor)



<http://www.youtube.com/c/MicrosoftReactor>



aka.ms/ReactorEmailSignUp



Reactor

We are constantly striving to create excellent content and would appreciate if you could take this brief survey.

Survey Link: <https://aka.ms/Reactor/Survey>

Please enter the event code 13071 at the start of survey

