

Poprvé s WSL

WSL (Windows Subsystem for Linux) je součást Windows umožňující používat běžné distribuce Linuxu (např. Ubuntu) v prostředí Windows. WSL 1 (od 2016) dovolilo spouštět v oknech běžících Windows linuxové textové aplikace, jejichž potřeby byly plněny emulací linuxového jádra. WSL 2 (od 2019) poskytuje s využitím virtualizačních schopností skutečné linuxové jádro a nabízí tak linuxovým aplikacím prakticky nativní prostředí a téměř plnou rychlost procesoru. Mimo jiné WSL 2 obsahuje interní X server a zprostředkuje tím bez dalšího běh grafických aplikací. Uživatelé a vývojáři tak získávají v prostředí Windows plnohodnotný přístup k možnostem Linuxu nedostatečně podporovaným ve Windows, jako jsou paralelizace pomocí MPI nebo vytěžování GPU pomocí Nvidia HPC SDK.

Instalace

► WSL lze provozovat od verze Windows 10. Nutná je aktivace hardwarové podpory virtualizace, sice běžně dostupné, ale často v BIOSu vypnuté (př. Asus: **Advanced-CPU Configuration-enable Intel Virtualization Technology**). Dále je třeba mít aktivované dvě potřebné Windows vlastnosti (v Lupě: **Turn Windows features on or off, enable Virtual Machine Platform, enable Windows Subsystem for Linux**), případně restartovat.

► Krátká cesta ke zpřístupnění aktuální verze WSL je pak řádkovým příkazem, zadaným v příkazovém okně (získaném např. v okénku Run po klávesové kombinaci **Win+R** spuštěním příkazu **cmd**):

wsl --install (na stroji bez předchozí instalace WSL)

wsl --update (na stroji s předchozí instalací WSL)

Pro jistotu lze stvrdit používání WSL 2 příkazem:

wsl --set-default-version 2

a vyžádat restart WSL:

wsl --shutdown

Pokud na stroji s dříve instalovaným WSL přežívá neaktuální distribuce Linuxu, je vhodná chvíle k její odinstalaci (v Lupě např.: zadat **Apps & features**, najít Ubuntu, zvolit Uninstall).

► Instalovat novou linuxovou distribuci (i více různých distribucí) je rychlé pomocí aplikace **Microsoft Store** (v Lupě zadat **Store**, v **Search** zadat Ubuntu a vybrat aktuální verzi, např. 22.04 LTS, zvolit **Get** a poté **Open**). Po řádově minutové akci se připravená distribuce ozve výzvou pro vložení nového uživatelského jména a hesla.

► Je vhodná chvíle nastavit přenositelnost obsahu schránky mezi Windows a WSL pomocí **Ctrl+Shift+C/V** kliknutím pravého tlačítka myši na horní liště příkazového okna a zapnutím příslušné volby jak v **Properties**, tak v **Defaults**.

► Stav WSL a linuxových distribucí lze pozorovat pomocí příkazů **wsl --version**; **wsl --status**; **wsl -l -v**.

První spuštění

► Instalované linuxové distribuce lze spustit výběrem položky ve Start menu, nebo příkazem **wsl** zadaným v okénku Run (po kombinaci **Win+R**) nebo v příkazovém okně (získaném příkazem **cmd** po **Win+R**) nebo při podobných příležitostech. Objeví se textová výzva linuxového shellu a lze si ještě znovu ověřit, že jsme ve WSL 2 a nikoliv 1; příkaz **uname -a** vrací (v květnu 2023):

(WSL 1) Linux box 4.4.0-19041-Microsoft #2311-Microsoft Tue Nov 08 17:09:00 PST 2022 x86_64 x86_64 x86_64 GNU/Linux

(WSL 2) Linux box 5.15.90.1-microsoft-standard-WSL2 #1 SMP Fri Jan 27 02:56:13 UTC 2023 x86_64 x86_64 x86_64 GNU/Linux

► Čerstvě instalovanou distribuci je žádoucí updatovat a přidat potřebné sady balíčků. To je třeba dělat z pozice superuživatele, jehož práva lze získat buď pro jeden příkaz formou **sudo command**, nebo pro více příkazů přepnutím **sudo -i** s návratem příkazem **logout** nebo kombinací **Ctrl+D**. Update (prověření) a upgrade (občerstvení) instalovaných balíčků se provede příkazy: **apt update**; **apt upgrade**. Další balíčky si uživatel vybírá podle potřeby, např. dotazem **apt list | grep numpy**, a instaluje příkazem např. **apt install python3-numpy**. Výčet u nás užívaných balíčků je připojen níže.

► Souborový systém WSL, zahrnující systémové soubory, aplikace a domácí adresáře, je virtuálním diskem přístupným uvnitř WSL, nikoliv však zvnějšku. Všechny interní disky Windows jsou ze strany WSL přístupné snadno – disk **C:** je připojen jako **/mnt/c**, disk **D:** je **/mnt/d** atd. Data lze tedy držet v adresářích dostupných jak z Windows, tak z WSL, bez potřeby přenosů. Je ovšem třeba vzít v úvahu, že práce se soubory ve WSL je výrazně rychlejší v oblasti virtuálního disku než ve společných adresářích připojených v **/mnt**. Na externí USB disky přistupovat lze také, ale prvotní konfigurace je (zvláště ve Windows 10) pracná.

► Příjemnou schopností WSL je, že v něm lze volat příkazy Windows, např. (ve WSL) **cmd.exe**, podobně jako ve Windows lze pomocí WSL volat příkazy Linuxu, např. (v **cmd**) **wsl uname -a**.

► Výčet u nás užívaných balíčků:

7zip bzip2 curl gawk mc screen screenie unrar unzip htop nvtop tofrodos caca-utils linux-tools-common linux-tools-generic cmake git git-doc
(GCC) gcc gfortran g++ cpp-doc gcc-11-locales g++-multilib g++-11-multilib gcc-11-doc gcc-multilib make autoconf automake libtool flex bison
gdb gcc-doc gcc-11-multilib gfortran-multilib gfortran-doc gfortran-11-multilib gfortran-11-doc libcoarrays-dev glibc-doc libgd-tools libstdc++-11-doc
(Open MPI) libopenmpi-dev libcoarrays-dev libcoarrays-openmpi-dev libcaf-openmpi-3

(Python) `python-is-python3 python3-pip python3-dev idle3 ipython3 jupyter cython3 python3-numpy python3-scipy python3-matplotlib python3-numba python3-sympy python3-pandas python3-sphinx python3-mpi4py python3-nose python3-pyproj libjs-mathjax python3-cartopy python3-mpltoolkits.basemap python3-owslib python3-gmsh python3-pygmsh`

(gnuplot aj.) `gnuplot imagemagick x11-xserver-utils x11-apps`

(příprava pro GMT) `build-essential cmake libcurl4-gnutls-dev libnetcdf-dev gdal-bin geos-bin libgdal-dev libfftw3-dev libpcre3-dev liblapack-dev libblas-dev libglib2.0-dev ghostscript pcre2-utils graphicsmagick ffmpeg xdg-utils graphicsmagick python3-sphinx`

(příprava pro Aspect) `hdfview hdf5-tools liblapack-dev petsc-dev libp4est-dev trilinos-all-dev libdeal.ii-dev`

(GCC v9, v12) `add-apt-repository ppa:ubuntu-toolchain-r/test; apt install gcc-9 gfortran-9 g++-9 gcc-12 gfortran-12 g++-12`

Grafické aplikace WSL a vzdálených strojů

► WSL 2 startuje v textovém okně, zahrnuje však X server a je schopno bez dalšího otevírat linuxové grafické aplikace v prostředí Windows. Příklad: `xclock`; `xeyes`; `gnuplot -e 'plot x**2; pause -1'`. Tuto schopnost poskytuje i pro vzdálené linuxové stroje, takže pokud se z Windows připojujeme nikoliv pomocí Windows ssh, ale pomocí WSL ssh, můžeme pak snadno užívat i grafické aplikace:

(Windows) `ssh -X user@machine` ... vyžaduje externí X serverovou aplikaci

(WSL) `ssh -X user@machine` ... používá interní X server

(Windows) `wsl ssh -X user@machine` ... používá interní X server

► Lze si tak ve Windows připravit dávkový soubor `novak.bat` pro připojování uživatelů například na stroj `novak`:

```
@wsl ssh -X %* josef@novak.domain
```

a používat ho pro uživatele `josef` voláním `novak` a pro jiné uživatele voláním například `novak -l karel`.

► Připojení ke vzdáleným strojům může být vhodné zajistit místo hesla pomocí SSH klíčů. V tom případě je třeba použít SSH klíče z WSL (tedy z WSL adresáře `~/.ssh`), nikoliv z Windows (tedy z `C:\Users\username\.ssh`). SSH klíče se ve WSL vytvoří běžným způsobem, například `ssh-keygen -t rsa` (s prázdnými odpověďmi na otázky), a obsah souboru `~/.ssh/id_rsa.pub` se přidá na vzdáleném stroji do souboru `~/.ssh/authorized_keys`.

► Pokud ssh příkaz neakceptuje krátká jména vzdálených strojů a vyžaduje plný tvar `machine.domain`, lze si pomoci uvedením jmen spolu s příslušnými IP čísly do souboru `/etc/hosts`. To lze opsat z nějakého už takto konfigurovaného stroje. (Mimo WSL je tato možnost také, soubor `hosts` je v adresáři `C:\Windows\System32\drivers\etc`.) Soubor `/etc/hosts` patří ve WSL k dynamicky generovaným souborům; po ručních změnách je třeba ho zafixovat přidáním řádků `[network] generateHosts=false` do souboru `/etc/wsl.conf`.

Poznámky k aplikacím ve WSL

► WSL umožňuje na stroji s Windows běh aplikací, které jsou omezeně nebo nejsou vůbec implementovány pro Windows, například MPI paralelizované úlohy nebo úlohy s numerickými knihovnami podporovanými v Linuxu.

► WSL umožňuje na stroji s Windows práci nástrojů Nvidia HPC SDK (neimplementovaných pro Windows), poskytujících například v překladačích C a Fortranu OpenACC direktivy pro přenos výpočtů na GPU. WSL a Ubuntu jsou přizpůsobeny tak, že umějí vytěžit Nvidia GPU s Windows driversy.

Odkazy

Wikipedia https://en.wikipedia.org/wiki/Windows_Subsystem_for_Linux

Microsoft <https://aka.ms/wsl>, <https://aka.ms/wsl2>

Ubuntu <https://wiki.ubuntu.com/WSL>

Nvidia <https://docs.nvidia.com/cuda/wsl-user-guide>