

Bash (Bourne Again SHell)

Bash je vylepšená verze původního Unixového shellu (sh). Byl vytvořen v rámci projektu [GNU](#) Brianem Foxem v roce 1989. Dnes je výchozím shellem pro většinu linuxových distribucí a klíčovým nástrojem pro správce systémů, vývojáře a [DevOps](#) inženýry.

1. Základní principy fungování

Bash funguje v cyklu **REPL** (Read-Eval-Print Loop):

- Read:** Přečte příkaz, který uživatel napsal.
- Eval:** Vyhodnotí příkaz (expanze proměnných, wildcards, hledání cesty k programu).
- Print:** Spustí program a vypíše výsledek na standardní výstup.
- Loop:** Čeká na další příkaz.

2. Klíčové funkce Bashe

Příkazová runda (Piping & Redirection)

Jednou z největších sil Bashe je schopnost propojovat jednoduché programy do složitých řetězců:

- Standardní výstup (STDOUT):** `ls > soubory.txt` (zapiše výpis do souboru).
- Roura (Pipe):** `cat data.txt | grep „Chyba“ | wc -l` (přečte soubor, najde řádky s chybou a spočítá je).

Expanze a Wildcards

- Hvězdička (*):** Zastupuje libovolný počet znaků (např. `rm *.log` smaže všechny logy).
- Dolar (\$):** Přístup k proměnným (např. `echo $USER`).

3. Programování v Bashi (Bash Scripting)

Bash skripty umožňují automatizovat opakující se činnosti. Skript je v podstatě textový soubor se seznamem příkazů.

Struktura skriptu

Každý skript by měl začínat tzv. **Shebangem**, který říká systému, jaký interpret použít:

```
#!/bin/bash
```

```
# Toto je komentář
echo "Zálohuji data..."
tar -czf zaloha.tar.gz /home/uzivatel/dokumenty
echo "Záloha hotova."
```

Logické konstrukce

Bash podporuje podmínky, cykly a funkce:

- **Podmínky:** `if [$a -gt $b]; then ... fi`
- **Cykly:** `for i in {1..5}; do echo $i; done`

4. Uživatelské přizpůsobení

Chování Bashe lze ovlivnit v konfiguračních souborech:

- **.bashrc:** Spouští se pro každý nový terminál. Zde se definují **aliasy** (zkratky).
- **.bash_profile:** Spouští se při přihlášení uživatele.

Příklad aliasu: `alias ll='ls -la -color=auto'`

5. Srovnání s jinými shelly

Shell	Charakteristika
sh	Původní Bourne shell, velmi jednoduchý, dnes se používá pro čistou kompatibilitu.
Bash	Standard, bohaté funkce, skvělý pro interaktivní práci i skripty.
Zsh	Modernější, populární na macOS, pokročilé doplňování příkazů (plugin Oh My Zsh).
Fish	Extrémně uživatelsky přívětivý, barevný, ale není plně kompatibilní s Bashem.

6. Proč je Bash nepostradatelný?

1. ****Správa serverů:**** Většina cloudových instancí se spravuje vzdáleně přes SSH právě pomocí Bashe.
2. ****[[CI/CD]] Pipelines:**** Automatizační skripty v GitLabu nebo GitHub Actions jsou často psány v Bashi.
3. ****Rychlost:**** Často je rychlejší napsat jednořádkový příkaz v Bashi než hledat funkci v grafickém rozhraní.

Tip pro začátečníky: Naučte se klávesovou zkratku **Tab**. Bash za vás automaticky doplní názvy souborů a příkazů, což dramaticky zrychluje práci.

Související: [Linux](#), [I/O](#), [DevOps](#), [Python](#)

From:

<https://serviceit.cz/> - **IT ENCYKLOPEDIE**

Permanent link:

<https://serviceit.cz/doku.php?id=bash>

Last update: **2025/12/31 18:09**

