

POSIX: Standard pro přenositelnost systémů

POSIX (zkratka pro *Portable Operating System Interface*, přičemž „X“ bylo přidáno pro „unixový nádech“) definuje rozhraní mezi aplikací a operačním systémem. Pokud je systém „POSIX-compliant“, znamená to, že program napsaný podle těchto pravidel se zkompileje a spustí na jakémkoli jiném systému splňujícím tento standard.

1. Historický kontext a vznik

V 80. letech existovalo mnoho verzí [Unixu](#) (System V, BSD, AIX atd.), které se od sebe mírně lišily. Vývojáři museli pro každou platformu kód upravovat, což bylo nákladné.

- **Richard Stallman:** Navrhl název POSIX na žádost IEEE.
- **Cíl:** Vytvořit společného jmenovatele pro systémová volání, správu procesů a práci se soubory.

2. Co POSIX definuje?

Standard pokrývá širokou škálu oblastí, které považujeme za samozřejmé, ale dříve byly nejednotné:

Systemová volání (System Calls)

Definuje, jak program žádá jádro o služby. Například:

- `open()`, `read()`, `write()`, `close()` pro práci se soubory.
- `fork()` a `exec()` pro vytváření a spouštění nových procesů.

Shell a utility

Určuje standardní chování příkazového řádku ([Bash](#), `sh`) a základních nástrojů jako `ls`, `grep`, `awk` nebo `sed`. To zaručuje, že skripty napsané na jednom systému poběží i na druhém.

Vlákna (Pthreads)

Standard **POSIX Threads** (Pthreads) definuje API pro vícevláknové programování. Umožňuje programům v [C++](#) nebo [C](#) využívat více jader [CPU](#) jednotným způsobem.

Regulární výrazy

Definuje syntaxi pro vyhledávání textu (Basic a Extended Regular Expressions), kterou používá většina unixových nástrojů.

3. Úrovně shody (Compliance)

- **Certifikované systémy:** Oficiálně splňují standard a zaplatily za certifikaci. Patří sem macOS, Solaris, AIX nebo HP-UX.
- **Většinou kompatibilní (Unix-like):** Linux a FreeBSD nejsou technicky vzato certifikovány (aby se ušetřilo za poplatky a zachovala svoboda vývoje), ale v praxi se standardu drží velmi přísně.
- **Ne-POSIX systémy:** Microsoft Windows není POSIXový systém, ale nabízí vrstvy jako WSL (Windows Subsystem for Linux), které kompatibilitu simulují.

4. Proč je POSIX důležitý pro vývojáře?

Díky POSIXu může vývojář použít například Clang k sestavení webového serveru na macOS a následně tentýž kód (bez změn v systémových voláních) nasadit na Linuxový server v cloudu.

Oblast	Výhoda POSIXu
Souborový systém	Cesta se vždy odděluje lomítkem (/).
Chybové stavy	Standardní chybové kódy jako EACCES (přístup odepřen).
Signály	Standardizované signály jako SIGKILL nebo SIGTERM.

Zajímavost: Ačkoliv se Windows snaží od POSIXu distancovat, jeho síťový stack a některé vnitřní mechanismy historicky vycházely z BSD kódu, aby bylo snazší portovat síťové aplikace.

Související: [Unix](#), [Linux](#), [FreeBSD](#), [Solaris](#), [Bash](#)

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

Permanent link:
<https://serviceit.cz/doku.php?id=posix>

Last update: 2025/12/31 18:14

