

Serverless Computing (Serverless)

Serverless computing (často zkráceně jen **Serverless**) je model provádění v cloud computingu, ve kterém poskytovatel cloudu plně spravuje spouštění kódu. Na rozdíl od tradičního PaaS se zde vývojář nestará ani o velikost instancí nebo škálování serverů – platforma přiděluje prostředky dynamicky v reálném čase.

Název „Serverless“ neznamena, že servery neexistují; znamená to, že pro vývojáře jsou zcela **abstrahovány**.

Rozdíl mezi PaaS a Serverless

I když jsou si tyto modely blízké, hlavní rozdíly spočívají v granulitě a způsobu účtování:

Vlastnost	PaaS (Platform as a Service)	Serverless (FaaS)
Škálování	Často manuální nebo na základě pravidel (CPU > 70 %).	Plně automatické a okamžité (podle počtu požadavků).
Účtování	Platba za běžící instanci (i když na ni nikdo nepřistupuje).	Platba pouze za dobu běhu kódu (milisekundy).
Životní cyklus	Aplikace běží neustále (Always-on).	Kód se spustí při události a po dokončení zanikne.
Abstrakce	Správa runtime a verze OS.	Pouze nahrání čisté funkce (logiky).

FaaS: Function as a Service

Jádrem serverless architektury je **FaaS**. Programátoři rozdělí svou aplikaci na malé, samostatné funkce, které jsou spouštěny specifickými **událostmi (Events)**.

Příklady událostí (Triggery):

- **HTTP požadavek:** Někdo klikne na odkaz nebo zavolá API.
- **Změna v databázi:** Do tabulky byl přidán nový řádek.
- **Nahrání souboru:** Do cloudového úložiště byl nahrán obrázek (spustí se funkce pro změnu velikosti).
- **Časový plán:** Funkce se spustí každou hodinu (Cron job).

Výhody Serverless modelu

- **Nulová správa:** Žádné aktualizace OS, žádné nastavování load balancerů.
- **Platba za využití (Zero to Hero):** Pokud aplikaci nikdo nepoužívá, náklady jsou doslova nulové.
- **Extrémní škálovatelnost:** Pokud přijde 10 000 požadavků najednou, platforma spustí 10 000 instancí funkce paralelně.

Nevýhody a omezení

- **Cold Start (Studený start):** Pokud funkce nebyla dlouho spuštěna, trvá několik milisekund (i sekund), než se inicializuje prostředí, což může způsobit latenci.
- **Limitace času:** Většina poskytovatelů omezuje maximální dobu běhu jedné funkce (např. na 15 minut).
- **Vendor Lock-in:** Přejechod mezi AWS Lambda a Azure Functions je náročný kvůli rozdílným API pro trigger.
- **Debugging:** Testování složitých serverless architektur je náročnější než u monolitických aplikací.

Populární Serverless platformy

- **AWS Lambda:** Průkopník a nejrozšířenější řešení.
- **Google Cloud Functions:** Silná integrace s ostatními službami Google Cloud.
- **Azure Functions:** Podpora mnoha jazyků a skvělá integrace s .NET prostředím.
- **Vercel / Netlify:** Populární mezi frontend vývojáři pro „Edge Functions“.

— **Související termíny:** [PaaS](#), [Cloud Events](#), [Mikroslužby](#), [API Gateway](#).

From:

<http://serviceit.cz/> - IT ENCYKLOPEDIA

Permanent link:

<http://serviceit.cz/doku.php?id=serverless>

Last update: **2026/01/03 15:08**

