

Rolling updates*

Silvia Kuchtová

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií
xkuchtovas@stuba.sk

18. december 2021

Abstrakt

Priebežná aktualizácia je široko používaná technika na aktualizáciu softvéru, pričom služba poskytovaná viacerými inštanciami softvéru zostáva dostupná. Aktualizácie softvéru v systémoch, a najmä v rozsiahlych systémoch webových služieb sú nevyhnutné a ich množstvo sa zvyšuje. Keď aktualizácia zlyhá, následky môžu byť vážne. Je to dokonca ťažšie, keď chceme zachovať dostupnosť a obsluhu systému zákazníkom počas aktualizácie. Problémy s takýmto dynamickými aktualizáciami softvéru sú dobre známe. V súčasnosti je bežnou praxou používanie priebežných aktualizácií, keď je služba poskytovaná viacerými inštanciami softvéru. V jednom okamihu sa pri priebežnej aktualizácii odstráni malý počet inšancií, ktoré používajú starú verziu softvéru a nahradí ich inštanciami v novej verzii. Tento proces sa opakuje, až kým nie sú všetky inštalácie v starej verzii nahradené.

1 Úvod

Tento článok vám pomôže lepšie pochopiť, čo sú to prebežné aktualizácie (rolling updates) a ako v skratke tieto aktualizácie fungujú. Tieto základné informácie si objasníme v častiach 2 a 3. V článku si spomenieme aj výhody a nevýhody, ktoré takéto aktualizácie prinášajú a v nakoniec to celé zhodnotíme v záverečnej časti 6.

2 Čo sú priebežné aktualizácie

Väčšina programov sa časom aktualizuje, zvyčajne prostredníctvom štandardného vydania aktualizácie. Pri štandardných aktualizáciách vývojár softvéru vytvorí úplne novú verziu programu a aktualizácie sa bežne objavujú každých niekoľko týždňov alebo mesiacov. Ak vývojár používa schému priebežného vydávania, postupuje sa inak. Namiesto zriedkavých aktualizácií sa aktualizácie bežne vykonávajú každý deň alebo každých niekoľko dní. Vývojár tiež pracuje len na aktualizácii jednej programovej vetvy, zatiaľ čo štandardné aktualizácie

*Semestrálny projekt v predmete Metódy inžinierskej práce, ak. rok 2021/22, vedenie: Ján Lučanský

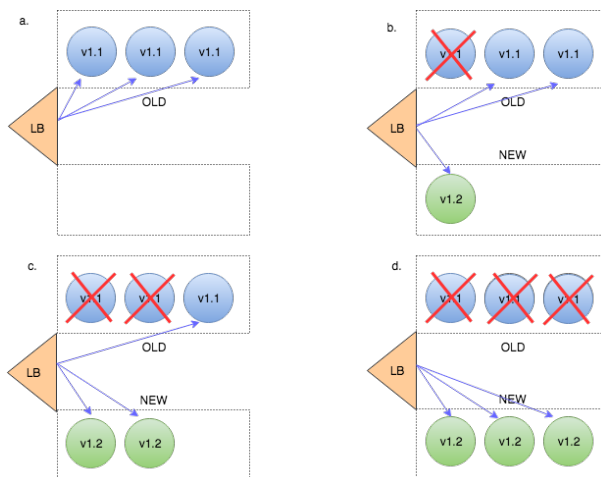
pracujú na viacerých vetvách.

Priebežné aktualizovanie je preto postup pri aktualizácii softvéru, ktorý namiesto vytvárania veľkých aktualizácií naraz zahŕňa mnoho priebežných aktualizácií. Výhodou tejto metódy je, že aktualizácie vychádzajú oveľa rýchlejšie a zvyčajne sa s nimi programátorom ľahšie pracuje. Zároveň však aktualizácie nemusia byť také dôkladné [3].

3 Ako priebežné aktualizácie fungujú

Pri tradičnej aktualizácii softvéru sa aplikačné servery odstavia, kým sa ich softvér aktualizuje a testuje, a potom sa opäť uvedú do prevádzky. To môže mať za následok značné prestoje aplikácie - najmä ak neočakávané chyby alebo problémy prinúti vývojára vrátiť inštaláciu na predchádzajúcu verziu.

Pri priebežnej aktualizácii je v danom čase mimo prevádzky len časť kapacity servera pre aplikáciu. Príkladom môže byť trojvrstvová aplikácia pozostávajúca z front end, back end a databázy, s tromi uzlami v každej vrstve. Každý z troch uzlov front-end aplikačného servera prijíma prevádzku cez **load balancer** (*vyrovnávač záťaže*). Pri tradičnej aktualizácii vyrovnávač záťaže uzavrie všetku prevádzku aplikácie, aby servery prešli do režimu offline a aktualizovali sa. Pri priebežnom nasadení môže organizácia vyradiť jeden z uzlov v každej vrstve z prevádzky, pričom vyrovnávač záťaže je nakonfigurovaný tak, aby presmeroval prevádzku na zostávajúce servery, na ktorých stále beží overená, aktuálna verzia softvéru. Nečinné servery prijímajú aktualizácie a testujú sa; zostávajúce servery v režime online podporujú prevádzku používateľov [2].



Obr. 1: Schéma postupu priebežných aktualizácií.
[1]

Skrátene povedané, pri priebežnej aktualizácii sa odstráni malý počet inštancií starej verzie a potom ich nahradí inštanciami novej verzie. Tento proces sa

opakuje, až kým sú všetky inštancie starej verzie nahradené [4]. Tento postup môžeme vidieť aj na predchádzajúsom obrázku 1

Priebežné aktualizácie môžu zahŕňať testovaciu fázu, v ktorej vyrovnávač záťaž smeruje len obmedzenú alebo testovaciu prevádzku, kým sa nový softvér nakonfiguruje a vyskúša. To nemá vplyv na používateľov, ktorí stále pristupujú k aplikácii na zostávajúcich serveroch, ktoré ešte neboli aktualizované [2].

Po úspešnom otestovaní softvéru sa server vráti do prevádzky a ďalší uzol servera sa odpojí, aby sa proces zopakoval, pričom sa pokračuje, kým nie sú všetky uzly servera v klastri aktualizované a beží na nich požadovaná verzia softvéru aplikácie. **Celá aplikácia bola nepretržite dostupná počas celého postupného aktualizovania.** [2]

4 Výhody a nevýhody

Jednou z hlavných výhod priebežného vydávania pre vývojára je, že zvyčajne môže aktualizácie vytvárať v malom časovom úseku. Aktualizovaný program bude tiež často fungovať lepšie. Program je neustále aktualizovaný, takže by mal zaznamenať vyššiu rýchlosť aplikácie a chyby by mali byť rýchlo odstránené.

Ďalšou výhodou priebežných nasadení je, že ich možno ľahko vrátiť späť, ak sa niečo pokazí. Ak v priebehu aktualizácie zaznamenáme problémy - zvyčajne je možné "vrátiť" postupné nasadenie a začať odstraňovať novú verziu aplikácie a znovu spustiť starú verziu.

Hoci má priebežné vydávanie programu svoje výhody, má aj niektoré nevýhody. Pri štandardných aktualizáciách má vývojár dostatok času na diagnostikovanie prípadných chýb alebo závažných problémov ovplyvňujúcich program. V schéme priebežných aktualizácií vývojár neustále vykonáva aktualizácie, takže si nemusí všimnúť závažné problémy. Je tiež menej času na testovanie aktualizácií, takže sa môžu vyskytnúť zjavné chyby, ktoré by sa pri štandardných aktualizáciách opravili. Program sa mení tak často, že aj keď sú zmeny malé, robia softvér zraniteľným voči vírusom a hackerským problémom.

Počas vykonávania priebežnej aktualizácie bude prostredie trochu nepredvídateľné, pretože v ňom súčasne bežia rôzne verzie aplikácie. A napokon, vyradenie serverov z prevádzky znamená, že počas aktualizácie je k dispozícii menšia kapacita pre prevádzku používateľov.

V rozsiahlych systémoch a aplikáciách je zapojených veľa inštancií, a preto priebežná aktualizácia trvá hodiny alebo aj oveľa dlhšie. Počas dlhej priebežnej aktualizácie sú zlyhania a v skutočnom dátovom centre môže obnova po poruche a údržba systému trvať hodiny alebo dni. [5]

5 Blue/green model

Blue/green deployment je model vydania aplikácie, ktorý postupne prenáša používateľskú prevádzku z predchádzajúcej verzie aplikácie na takmer identickú novú verziu.

Stará verzia sa môže nazývať modré prostredie, zatiaľ čo nová verzia sa môže označovať ako zelené prostredie. Po úplnom prenose produkčnej prevádzky z modrého do zeleného prostredia môže byť modré prostredie v pohotovostnom režime pre prípad vrátenia alebo stiahnutia z produkcie a aktualizácie, aby sa stalo šablónou, na základe ktorej sa vykoná ďalšia aktualizácia.

6 Záver

V tomto článku sme predstavili priebežné aktualizácie, funkciu, ktorá je užitočná, pretože umožňuje aktualizovať bežiacie aplikácie na novšie bez prerušenia prevádzky. Tento model však má svoje nedostatky, z ktorých by pár mohlo byť v budúcnosti odstránených. Zamedzením chýb by mohlo byť aj zavedenie dlhšej testovacej fázy alebo vykonávanie aktualizácií v dlhšom časovom úseky, pre lepšiu kontrolu chybovosti novej verzie.

Literatúra

- [1] <https://codefresh.io/kubernetes-tutorial/continuous-deployment-strategies-kubernetes-2/>.
- [2] Stephen J. Bigelow. Rolling deployment. <https://searchitoperations.techtarget.com/definition/rolling-deployment>.
- [3] Alex Newth. What is a rolling release? <https://www.easytechjunkie.com/what-is-a-rolling-release.htm>.
- [4] Daniel Sun, Len Bass, Alan Fekete, Vincent Gramoli, An Binh Tran, Sherry Xu, and Liming Zhu. Quantifying failure risk of version switch for rolling upgrade on clouds. In *2014 IEEE Fourth International Conference on Big Data and Cloud Computing*, pages 175–182, 2014.
- [5] Daniel Sun, Alan Fekete, Vincent Gramoli, Guoqiang Li, Xiwei Xu, and Liming Zhu. R_isup_i2i/sup_ic: Robust rolling-upgrade in clouds. *IEEE Transactions on Dependable and Secure Computing*, 15(5):811–823, 2018.