

Az infrastruktúra menedzsment a szoftverfejlesztés egyik kritikus és komplex részfeladata. Manapság a legtöbb szoftvercég felhő alapú megoldásokat használ ezen részprobléma leküzdésére. Új felhő-erőforrások létrehozása és meglévők újrakonfigurálása mindennapi feladat ezen cégek számára. Ezek a folyamatok meglehetősen komplexek és a menedzselt infrastruktúra méretével a komplexitás gyorsan növekszik. Az erőforrás-foglalás és -újrakonfigurálás során elkövetett nagyobb hibák könnyen vezethetnek részleges vagy akár teljes szolgáltatás-kieséshez is.

A felhő alapú erőforrások kezelésére több módszer is létezik. Az egyik ilyen technika az "Infrastructure as Code", röviden IaC. Az IaC megoldások deklaratív, automatizált módszert nyújtanak felhő-erőforrások kezelésére.

Viszont az olyan automatizált felhő-erőforráskezelők megjelenésével, amelyeknek jelentős kihatása van a céges infrastruktúra állapotára és ezáltal a termék minőségére és költségeire, megnő az igény az erőforráskezelés lépéseinek verifikációjára.

Ebben a dolgozatban az IaC megoldások és az IaC erőforráskezelés lépéseinek automatikus verifikációjára koncentrálnunk. A mai napig az IaC által végzendő műveletek ellenőrzését mérnökök végzik. Mi egy olyan megoldást dolgoztunk ki ami ezeket a mérnököket segíti a verifikációs folyamatban, ezáltal elősegítheti az emberi hibák számának csökkenését.

A dolgozatban bemutatjuk a felhő-alapú erőforrásoknak és azok kezelésének az alapvető koncepcióit. Valamint az eszközöket és módszereket amiket a dolgozat során felhasználtunk. Feltesszük, hogy az olvasó rendelkezik alapvető tudással a webszolgáltatások infrastruktúrájára vonatkozóan, de részletesen bemutatjuk a felhő felépítését és az IaC alapú erőforrás kezelést.

Ezen felül megvitatunk néhány - jelenleg létező - megoldást, amelyek a mérnökök munkáját támogatják a felhők biztonságosabb újrakonfigurálásának érdekében.

Ezek után bemutatjuk új algoritmusunkat és implementációját, amelyet az IaC újrakonfigurálásoknak egy speciális, webszolgáltatás specifikus, részhalmazának verifikációjára fejlesztettünk ki. Majd értékeljük képességeit és gyengeségeit egy esettanulmány során, ahol egy apró hibát fogunk injektálni egy IaC leírásba, amelyet könnyű szem elől téveszteni, azonban jelentős hatással van a webszolgáltatás elérhetőségére.

Az esettanulmányt követően részletezzük a jövőbeli terveinket, hogy mindennapi mérnöki használatra alkalmas állapotra fejlesszük a jelenlegi proof-of-concept projektünket, illetve előre vetítsük hosszú távú céljainkat az algoritmus további fejlesztésére vonatkozóan.