DEVOPS

2. forduló

<epam>

A kategória támogatója: EPAM

Ismertető a feladathoz

Útmutató:

- A **radio button-os kérdésekre** egy helyes válasz van.
- **Ha lejár a feladatlap ideje, a rendszer AUTOMATIKUSAN** beküldi azt az addig megjelölt válaszokkal.
- Az **adatbekérős feladatokra NEM jár részpontszám**, csak a feleletválasztósakra.
- **Badge-ket** a 4.forduló után kapsz majd először.
- Az **adatbekérős kérdéseknél** igyekeztünk minden variációt megadni (kisbetű, nagybetű, szóköz), de ha mégis eltérést tapasztalsz a megoldásokban, kérjük, jelezd felénk!

+1: Azért szólunk, hogy senkit ne a végén érjen meglepetés: a játék nem tipp-mix és csapatkategória sincs! Természetesen akinek nem inge...

Jó versenyzést kívánunk!

Felhasznált idő: 06:57/10:00 **Elért pontszám:** 4/8

1. feladat 0/1 pont

Sajnos jogi szabályozások miatt szükséges, hogy bármilyen, az éles rendszerbe kikerülő új szoftververzió egy többlépcsős manuális jóváhagyáson essen át, amely esetenként napokig is eltarthat.

Milyen megközelítést választanál, hogy a release folyamat megfeleljen a szabályozásnak és a lehető legtöbb lépés automatizálható legyen?

Válasz

- ○ Continous Integration, mivel biztosítja, hogy a fejlesztők gyorsan, jó minőségben készítsék el az új szoftververziókat, majd ezek telepítése a jóváhagyás után manuálisan történne, így garantálva a maximális kontrolt afelett, hogy mi kerül ki az éles rendszerbe.
- ● Continous Integration és Continous Deployment, amely során a fejlesztők a code merge előtt beszerzik a szükséges jogi jóváhagyást, így garantálva, hogy az automatizált lépések során a kód minősége megfelelő és megfelel a szabályozásnak is.
  Ez a válasz helytelen, de megjelölted.
- ● Continous Integration és Continous Delivery, amely igy lehetőséget biztosít bármely változás telepítésére, azonban maga a telepítés nem teljesen automatikus, így megoldható egy manuális jóváhagyási lépés integrálása is a folyamatba.
  Ez a válasz helyes, de nem jelölted meg.

- Continous Integration és Canary Deployment segítségével lehetne biztosítani a kód minőségét, valamint hiba esetén a gyors visszaállást, így előzve meg a komoly üzleti, valamint presztizs beli károkat.

Magyarázat

Continous Integration és Continous Delivery garantálja, hogy bármely változtatás, amely átment a szükséges teszteken az telepíthető és megfelelően használható, azonban magának az adott változásnak a telepítését nem hajta kötelezően végre, így egyéb feltételek, pl. egy emberi jóváhagyás is beépíthető a rendszerbe.

2. feladat 0/1 pont

Felmerült a gyanú, hogy az általad felügyelt AWS accountot feltörték. Melyik szolgáltatás segítségével próbálnád kideríteni, hogy a támadók milyen API hívásokat hajthattak végre, és mit változtattak meg?

Válasz

- CloudWatch
  Ez a válasz helytelen, de megjelölted.
- CloudTrail
  Ez a válasz helyes, de nem jelölted meg.
- Config
- Trusted Advisor

Magyarázat

A CloudTrail szolgáltatás szolgál az AWS API hívások tárolására segítve ezzel a biztonsági ellenőrzéseket.

3. feladat 1/1 pont

Mi ennek a Pythonban írt függvénynek a visszatérési értéke, ha x = 4?

```
def f(x): return sum(range(x))
```

Válasz

- 4
- 6
  Ez a válasz helyes, és meg is jelölted.
- 91
- 12

Magyarázat

Mivel a range 0-től (x-1)-ig terjed, ezért az összeg 6 lesz.

4. feladat 0/1 pont

Egy standup során vérre menő vita alakul ki a git-ben a **force push** használatáról. Mi lehet az ellenérv a használatával kapcsolatban, ami miatt adott helyzetben inkább kerülni kéne?

Válasz

- (●) Inkonzisztenciát idézhet elő a git belső adatstruktúrájában.
  Ez a válasz helytelen, de megjelölted.
- ( ) Mások által klónozott repository-t a jóváhagyásuk nélkül változtat.
- (●) Mások nem tudják automatikusan beolvasztani (merge) a "force push"-olt ágon (branch) lévő új commit-okat a saját helyi követőágukra (tracking branch).
  Ez a válasz helyes, de nem jelölted meg.
- ( ) A push eseményre kötött hook-ok figyelmen kívül maradnak, az általuk végrehajtott biztonsági ellenőrzésekkel együtt.

Magyarázat

- A "force push" művelet a kód-ág végét (tip of branch) egy olyan commit-ra állítja, aminek nem minden commit őse, ami őse volt a force push előtt; így aki a force push előtt lemásolta a branchet, nem tudja fast-forward módon beolvasztani azt.
- Magasszintű git parancsok (pl. push, commit, ...) nem ronthatják el a belső adatstruktúrát.
- A git nem nyújt lehetőséget más felhasználók helyi repójának módosítására beleegyezésük nélkül.
- Az esemény hook-ok (event hooks) force push mellett is lefutnak.

5. feladat 1/1 pont

Mikor törlődnek a régi fájlok egy git repository-ból?

Válasz

- ( ) Alapértelmezetten 5 év után.
- ( ) Amikor a szemétgyűjtő rutin (garbage collector, git gc) fut és akkor az aktuális paraméterei szerint töröl.
- (●) Soha.
  Ez a válasz helyes, és meg is jelölted.
- ( ) A git nem töröl fájlokat kor alapján, hanem objektumokat, amik csak régi fájlok felépítéséhez kellenek.

Magyarázat

Soha, hacsak a felhasználó nem módosítja a git saját belső fájlain keresztül a commitokat visszamenőlegesen.

6. feladat 1/1 pont

Egy Terraformmal menedzselt infrastruktúra esetében hibakeresési okokból szükséged van arra, hogy néhány objektum állapotát ne módosítsa a kód futtatása, még akkor se, ha eltérés van az elvárt és az aktuális állapot között. Ezt melyik megoldással tudnád megtenni, ha megtehető egyáltalán?

Válasz

- (●) ignore_changes attribútum használatával megoldható

Ez a válasz helyes, és meg is jelölted.

- [ ] Nincs ilyen, ez a Terraform feladata, hogy a kódban leírt állapotot próbálja meg létrehozni
- [ ] Úgy állítod be a Terraformot futtató felhasználó jogosultságait, hogy az ne tudja megváltoztatni az adott objektum(ok) állapotát
- [ ] Nem oldható meg egyszerűen, azonban ha elmented az objektum(ok) állapotát, majd futtatás után mentésből visszaállítod, akkor elérheted az, hogy ne változzon meg minden

Magyarázat

A lifecycle blokkon belül az ignore_changes-t tudjuk erre használni. Hasznos lehet ez például olyan esetekben, hogy ha például egy Terraformon kívüli például compliance folyamat változtatja ezeket egy másik szabályrendszere alapján.

7. feladat 1/1 pont

Egy konténert nem sikerül leállítani a **SIGTERM** signál segítségével. Milyen egyéb lehetőséged van arra, hogy parancssorból leállítsd?

Válaszok

- [ ] docker terminate <container_id>
- [ ] docker rm <container_id>
- [x] docker stop <container_id>
  Ez a válasz helyes, de nem jelölted meg.
- [x] docker kill <container_id>
  Ez a válasz helyes, és meg is jelölted.

Magyarázat

- kill - A kill kapcsoló állítja le a konténert erőszakosan, SIGKILL signal küldéssel.
- terminate -vnem létezik
- rm -a már megállított konténerek törléséhez használható
- stop - a stop kapcsoló megállítja meg a futó konténert SIGTERM signal küldéssel

8. feladat 0/1 pont

Úgy döntötök, hogy az AWS-ben futó EC2 instancok alap metrikát Prometheus Node exporterrel szeretnétek összegyűjteni, használva annak alapértelmezett beállításait. Hogy állítanád be az EC2-höz kapcsolt security groupon a szabályokat, hogy megfelelően működjön a monitorozás, illetve teljesítsd a vállalat szigorú hálózatbiztonsági szabályait (least privileges) is?

Válasz

- [ ] Minden forrásból a bejövő kapcsolatok engedélyezése a TCP 9100 porton, így biztosítva, hogy a Prometheus szerver elérje a a node exportert, abban az esetben is, ha több szervert használtok.
- [x] Nem szükséges bejövő kapcsolat engedélyezése, mivel a node exporter automatikusan tölti fel az adatokat a megadott időközönként.
  Ez a válasz helytelen, de megjelölted.
- [x] Egyik válasszal sem biztosítható a monitorozás működése és a biztonsági szabályok betartása.
  Ez a válasz helyes, de nem jelölted meg.
- [ ] Security grouppal nem megoldható, azonban az AWS által bizotsított Network ACL-el (NACL), szabályozható a forgalom. Alapesetben a NACL nem engedélyez semmilyen külső forgalmat, így ennek alkalmazása biztosítja, hogy megfeleljünk a

hálózatbiztonsági irányelveknek.

Magyarázat

- A 9100 port engedélyezés bármely forrásból nem felel meg a szigorú szabályozásnak, mivel így bárki elérheti az adott EC2-n futó node exportert.
- Prometheus alapértelmezetten pull módban működik, így a szerver felől szükséges a bejövő kapcsolat engedélyezése
- Security grouppla beállítható a megfelelő szabályrendszer