

## DevOps Interns პრაქტიკული დავალებები

### გიორგი დონდოლაძე

#### 1) სერვერზე სტორიჯის გაზრდის პროცედურა.

ვთქვათ რომ გვაქვს სერვერი Centos 7 vCenter ვირტუალურ გარემოში , რომელზეც მიერთებულია 100GB იანი სტორიჯი , აღწერეთ დისკის გაზრდის პროცესი ისე , რომ სერვერის დარესტარტება არ დაგჭირდეს. ასევე რამოდენიმე ვარიანტის დაწერის შემთხვევაში მიუთითოთ პლიუსები და მინუსები.

სერვერზე უნდა იყოს ორი დისკი, ერთზე გაშვებული უნდა იყოს ოპერატიული სისტემა, მეორეზე ბაზა, რესტარტის გარეშე გაზრდისას პირველ რიგში, VM დან edit setting ში, virtual hardware ში ვზრდით hard disk 2 ის მოცულობას, შემდეგ რუთიდან ვშვებთ ბრძანებას - `echo 1 > /sys/class/block/sdb/device/rescan`

ვიყენებთ growpart რომელიც არის ნაწილი cloud-utils-package-ს ამიტომ ვაინსტალირებთ `sudo apt-get install -y cloud-utils` და ვუშვებთ `growpart /dev/sdb 1` და `resize2fs /dev/sdb1` ბრძანებებს,

#### 2) კომპანიის ოფისში არის 150-დე პერსონალური კომპიუტერი , სადაც არსებობს საჭიროება ოპერაციული სისტემის Windows ის გადაყენების , გთხოვს აღწეროთ ამ მიზნის მიღწევის ყველაზე სწრაფი/ოპტიმალური გზა.

ყველაზე ოპტიმალური გზა იქნება serva -ს გამოყენება, პირველ რიგში გასასწორებელი იქნება სერვას კონფიგურაცია, ვაკოპირებთ საინსტალაციო ფაილებს და ვსტარტავთ, შემდეგ ვაკოპირებთ ქსელის ბარათის დრაივერებს რომელ კომპიუტერებზეც გვინდა გადაყენება და ვაზიარებთ WIA\_WDS ფაილს ქსელში, თავიდან ვსტარტავთ ვერსას, ვრწმუნდებით რომ ბიოსში networking booting არის enabled-ზე და ვაინსტალირებთ ვინოდუსს.

#### 3) კომპანიას გააჩნია ოპერაციული სისტემა ( RHEL 7 ) სადაც არის დაპოსტილი Java აპლიკაცია , რომელიც იყენებს PostgreSQL მონაცემთა ბაზა, აპლიკაციამ რაღაც მომენტში დაიწყო შენელებულად მუშობა , გთხოვთ აღწეროთ ის თანმიმდევრული ნაბიჯები რომლითაც გამოიკვლევთ პრობლემას როგორც აპლიკაციის ასევე მონაცემთა ბაზის მხარეს.

1. ვამოწმებთ `pg_stat_activity`
2. შეიძლება იყოს რაიმე რიგი რაც აყოვესებს პროცესს
3. ვამოწმებთ ლოგებს
4. ვამოწმებთ მეხსიერების დისკს

#### 4) კომპანიას გააჩნია ბექაფების შესანახი სისტემა სადაც ყოველდღიურად უნდა იწერებოდეს პროდაქშენ სისტემების ბექაფები , საჭიროა რომ ბექაფების სისტემის

სტორიჯზე ჩაიწეროს მონაცემები Linux ოპერაციული სისტემიდან , როგორ შეგვიძლია მსგავსი სტორიჯის მიერთება ოპერაციულ სისტემაზე .

- 5) კომპანიას აქვს საჭიროება რომ განათავსოს Static საიტი ქლაუდში (AWS) , საიტს არ ექნება საკუთარი მონაცემთა ბაზა . ასევე საჭიროა უსაფრთხოების დეტალების გათვალისწინება , როგორ და რა პრინციპით დავნერგავდით მსგავს საიტს? (საიტი უნდა იყოს ხელმისაწვდომი ინტერნეტიდან.)
- 6) აპლიკაცია რომელიც დაწერილია .Net Core ზე იყენებს Postgre SQL მონაცემთა ბაზას , სადაც ხშირად ხდება დიდი რაოდენობით მონაცემთა ჩაწერა ლოგირების ცხრილებში , არსებობს საჭიროება რომ ეს ცხრილები პერიოდულად სუბთავდებოდეს , როგორც გავაკეთებდით ამას.

ამისთვის ძირითადად ორი მიდგომის გამოყენება შეიძლება: პერიოდული პროცესით წავშალოთ გარკვეული ჩანაწერები ბაზიდან ან ჩანაწერები დავაარქივოთ. ამ ყველაფრის მიღწევა შეიძლება როგორც .NET Core-ის აპლიკაციიდან (ქრონჯობების ან სხვა სახის ბექგრაუნდ სერვისებით/ტასკებით), ასევე ტექნიკურად შეიძლება მსგავსი scheduled job ბაზის მხარესაც გვექონდეს (Postgre SQL-ის ახალ ვერსიებს აქვს pg cron ექსტენშენის მხარდაჭერა), მაგრამ უკანასკნელი მეთოდი არც თუ კარგი გადაწყვეტაა.

- 7) კომპანია იყენებს AWS-ქლაუდს , სადაც საჭიროებს State Less სერვისებზე ტრაფიკის გადანაწილებას რომ ისინი თანაბრად იყვნენ დატვირთულნი , ასევე საჭიროა რომ ტრაფიკის ზრდის შემთხვევაში ავტომატურად დაემატოს ახალი სერვერ , როგორ შეგვიძლია მსგავსი რამის გაკეთება.

AWS ქლაუდი გვთავაზობს თავის ლოად ბალანსერს და აუტო სკეილინგის საშუალებებს. Auto Scaling-ისთვის AWS-ზე არსებობს არის Amazon EC2 ჯგუფები, რომლებსაც შეიძლება მარტივად „მიებას“ Elastic Load Balancer. არსებობს ოთხი ძირითადი ლოად ბალანსერის ტიპი: აპლიკაციის დონეზე, ნეტვეორქის დონეზე, გეითვეის დონეზე და კლასიკური.

- 8) Linux ოპერაციულ სისტემაზე არსებობს საჭიროება რომ კონკრეტული ექსექუტებალი გაეშვას ოპერაციული სისტემის ჩატვირთვის დროს ავტომატურა , როგორ გააკეთებდით მსგავს რამეს.

ვუშვებთ ბრძანებას ლინუქსის ტერმინალში:

```
sudo nano /etc/systemd/system/" SERVICE_NAME ".service
```

ვაკოპირებთ ბრძანებას და ctrl + x და შემდეგ y ვასეივებთ.

Description=" SERVICE\_DESCRIPTION"

Wants=network.target

After=syslog.target network-online.target

[Service]

Type=simple

ExecStart="COMMAND"

Restart=on-failure

RestartSec=10

KillMode=process

[Install]

WantedBy=multi-user.target

ვტვირთავთ სერვისს `sudo systemctl daemon-reload`

ვაენებლებთ სერვის ბრძანებით `sudo systemctl enable „SERVICE_NAME“`

ვსტარტავთ `sudo systemctl start „SERVICE_NAME“`

ვაშოვნებთ სტატუსს `systemctl status „SERVICE_NAME“` და ბოლოს ვტვირთავთ.

- 9) რა მთავარი განსხვავებებია Linux / Unix და Windows ოპერაციულ სისტემებს შორის. (გთხოვთ ჩამოთვალოს თითოეულის პლიუსები და მინუსებიც)

ლინუქსი არის უფასო unix os, მთავარი განსხვავება ვინდოუსთან შედარებით ფაილური სტრუქტურაა, ლინუქსი არის საკმაოდ ჩქარი OS რომელსაც შეუძლია დიდი წარმადობით იმუშაოს დაბალი მონაცემების მოწყობილობაზე, ასევე ლინუქსის პლიუსია უსაფრთხოება, თუ კი ვმუშაობთ სისტემაში როგორს მომხმარებელი და არა root-ი. ლინუქსში შესაძლებელია იმუშაოს ერთდროულად რამდენიმე ასეულმა პროგრამამ და ასევე ბევრმა მომხმარებელმა, თითოეულ მომხმარებელს საშუალება აქვს ერთდროულად იმუშაოს სხვადასხვა კონსოლში.

ვინდოუსის გამოყენება ლინუქსთან შედარებით მარტივია, მომხმარებლისთვის ინტერფეისი აღქმადი და კომფორტულია, ამიტომაც უფრო გავრცელებული და მოხმარებადია, მაგრამ იგი ფასიანი OS-ა, ის დაუცველია ვირუსებისა და მავნე პროგრამების შეტევების მიმართ. ასევე პერიოდულად საჭიროებს გადაყენებას შენელების გამო.

- 10) კომპიუტერული ქსელის მე -2 დონეზე , რომელი პროტოკოლი არსებობს რომელიც პასუხისმგებელია სვიჩირებად ქსელში Loop ებისგან თავის დაცვაზე , ასევე გთხოვთ აღწეროთ რა პრინციპით მუშაობს ეს პროტოკოლი.

ლუპებისგან თავის დასაცვაზე პასუხისმგებელია STP spanning tree protocol.

STP უზრუნველყოფს რომ ყველა მიმღებს და გამგზავნს შორის იყოს მხოლოდ ერთი root bridge და ბლოკავს სხვას რომ თავიდან აირიდოს loop-ი. სხვა bridge იბლოკება პორტიდან გაგზავნის თანავე. თუ კი root bridge-ს შეექმნება პრობლემა STP თავიდან გამოთვლის და საჭიროების შემთხვევაში განბლოკავს დაბლოკილ ალტერნატიულ გზას.