Linux-101 Advanced Project

DevOps'un günlük yaşamında karşılaştığı sorunlar ve DevOps mühendisinin projedeki dört görevi.

First\_Part çözümleri

Serdar adlı kullanıcının terminate ettiği instance id’lerini listelememiz ve result.txt dosyasına bu id’leri kaydetmemiz isteniyor.

1) cat event\_history.csv | grep -i "serdar" | grep -i "terminate" | grep -i "i-.\{17\}" | sort | uniq | tee result.txt

2) grep -i "serdar" event\_history.csv | grep -i "terminate" | grep -Eo "i-.{17}" | sort | uniq > result.txt

3) grep -i "serdar" event\_history.csv | grep -i "terminate" | grep -Eo "i-[a-zA-Z0-9]{17}" | sort | uniq > result.txt

4) cat event\_history.csv | grep -i "serdar" | grep -i "terminate" | grep -Eo "i-\w{17}" | sort | uniq | tee result.txt

5) grep “serdar” ./event\_history.csv | grep –i “terminateınstances” | egrep –o ‘\bi-0\w\*’ | sort | uniq | tee result.txt

6) awk –F, ‘/serdar**/ {print $1, $4, $10}’** event\_history.csv | grep “terminate\*” | grep “i-\*” | cut –d “:” –f2 result.txt | tee resultıd.txt

**(1. 4. 10 satırları al)**

(Farklı çözüm bulan var mı?)

Second\_Part çözümleri

E-ticaret şirketinde Sistem Yöneticisi olarak çalışıyorsunuz, işe alınan tüm insanlar için yeni Linux hesabı oluşturabilecek bir script yazmalısınız, "user\_password.sh" adlı bash script dosyası oluşturacağız.

1. Kullanıcı oluşturmalıyız bunun için root olmam gerek ya da root yetkisine ihtiyacım var Sudo ya ihtiyacımız var
2. ec2’deyim kullanıcı id mi (User ıd (UID)) kontrol ediyorum

-**Echo $UID** komutunu koşuyorum cevap 1000

-**Sudo su** komutunu koşuyorum ardından **whoami** komutu ile hangi kullanıcı olduğumu kontrol ediyorom cevap root.

- **Echo $UID** komutunu koşuyorum cevap 0 çünkü rootun UID’si sıfır

-exit ile çıkıyorum ec2’deyim

1. Scripti yazabilirz. (Unutmayalım çalışması için **chmod +x user\_password.sh** komutunu koşmalıyız)

**# !/bin/bash**

**if [[ ${UID} -ne 0 ]]**

**then**

**echo "please run this script as root or with superuser privileges."**

**exit 1**

**fi**

-Eğer UID sıfıra eşit değilse (rakam için –ne(eşit değil),-eq(eşit)) "lütfen bu scripti root olarak veya süper kullanıcı ayrıcalıklarıyla çalıştırın." desin ve exit “**1”** hata kodunu versin ve çıksın.

**(Ek bilgi, text düzenleyici komut: sed –i –e ‘s/\r$//’ user\_passwd.sh)**

-Kullanıcı adını nasıl alabiliriz read-p ile

**read -p "Enter your username: " USER\_NAME**

-kullanıcı adını al (giriş yap)

**read -p "Enter your comment: " COMMENT**

- (açıklama alanının içeriği)

**read -sp "Enter your password: " PASSWORD**

-şifreyi al (sp yazılanı ekranda göstermemesi için)

**useradd -c ${COMMENT} -m ${USER-NAME} 2>/dev/null**

-Hesabı oluşturun. (useradd komutu ile -c (commenti eklemek için) –m (bu kişi için home klasör oluştur) 2>/dev/null (eror mesajlarını çöpe at demek yazmasakda çalışır.))

**if [[ ${?} -ne 0 ]]**

**then**

**echo -e "\nThis username is already exist. Please select different username."**

**exit 1**

**fi**

-useradd komutunun başarılı olup olmadığını kontrol ediyoruz.Çünkü oluşturulmamış bir hesabın oluşturulduğunu kullanıcıya söylemek istemiyoruz. (-e :edit, $ {?} : bir önceki komut çalıştı mı? \n alt satıra inmesi için)

**passwd -e ${USER\_NAME} 1>/dev/null**

-Parolayı ayarlayın**(-e expire passwrdü ilk girişte değiştir** (passwd –help))

**if [[ ${?} -ne 0 ]]**

**then**

**echo -e "\nThe password for the accont could not set."**

**exit 1**

**fi**

-passwd komutunun başarılı olup olmadığını kontrol edelim.

**echo**

**echo -e "username: ${USER\_NAME}\n"**

**echo "Your password: ${PASSWORD}"**

-Kullanıcı adını, şifreyi ve kullanıcının oluşturulduğu görüntüleyin.

1. **ls –al** komutunu koşup user\_passwd.sh çalışabilir olduğunu görelim
2. **./user\_passwd.sh** dediğimizde bize scripti root olarak çalıştır uyarısını verecek
3. **sudo ./user\_passwd.sh**
4. **echo $?** Komutu ile komutumuzun çalışıp çalışmadığını kontrol edebiliriz. (0 çalıştı)
5. **cat etc/passwd** ile kullanıcıları listeleyelim
6. **sudo userdel Rose** adlı kullanıcıyı sildik

Forth\_Part çözümleri

Healthcare Company'de DevOps mühendisi olarak çalışıyorsunuz. 3 farklı görevin var

Task1: EC2 bulut sunucusunun özel IP'sini ec2 açıklama dosyasından ('info.json') alın ve 'terraform.tf' dosyasında bununla 'ec2\_private\_ip' değerini değiştirin. Linux komutlarıyla

Task2: Ansible playbook dosyası yazıyorsun, tek satırlı biçimde olan "certificate.pem" anahtar doyasını çok satırlı biçimini "new.pem" dosyasında oluşturun.

Task3: Sunucunuza ssh bağlantı günlükleri olan denetim günlüğü dosyanız var. Ancak bazı geçersiz kullanıcılar sisteminize bağlanmaya çalışır. Adlarını ve sisteminize kaç kez saldırmaya çalıştıklarını bulmak için size bir görev verildi. Lütfen onu bulun ve "invalid\_user.sh" dosyasına koyun.

Task1: info.jsondaki privat 20. Satırdaki ıp alıp terroform.tf 40. Satıra yapıştıracağım info.jsondaki privat ıp grep ile çekeceğim.

Cd ile Fort\_Partha ve oradan task1 e gelelim

**Çözüm1: grep “PrivateIpAddress” info.json | head -n1 | cut –d ‘ “ ‘ –f 4**

-istediğim ip adresini aldım ve şimdi terroform.tf ile değiştireceğim. Sed ile yapacağım.

**Sed –i “s/ec2-private\_ip/$(** **grep “PrivateIpAddress” info.json | head -n1 | cut –d ‘ “ ‘ –f 4)/g” terraform.tf**

-$() dedim çünkü istediğim komutu içine yazdığımda çalıştıracak, g (global), -i (üzerine yaz demek).

**Çözüm2: # !/bin/bash**

**İp=$(cat info.json | grep –m1 PrivateIpAddress | cut –d” : “ –f2 | cut –d’ “ ‘ –f2)**

**Sudo Sed –i “s/ec2-Private\_Ip/ $İp/” terraform.tf**

**Çözüm3: # !/bin/bash**

**İp=$(awk –F: ‘/PrivateIp/ {print $2}’ info.json | head -1 | tr –d ‘ “, ‘)**

**Sed –i “s/ec2-PrivateIp/ $İp/” terraform.tf**

Task2: cd .. cd task2

\n alt satıra geç demek \n leri nasıl aktif hala getirebiliriz?

echo “ab\ncd\nef” komutunu koştuğumda sonuç ab\ncd\nef dir fakat

echo -e “ab\ncd\nef komutunu koştuğumda sonuç ab

cd

ef

**e ile \ aktif hale getirebiliriz.**

**Çözüm1: cat certificate.pem**

**echo –e $(cat certificate.pem)**

\n ler aktif hale geldi ve pem dosyası multiline oldu.Bu sonucu new.pem e yazdırmak istiyorum

**echo –e $(cat certificate.pem) > new.pem**

**cat new.pem**

**Çözüm2: script dosyası oluşturalım realpem.sh içerisine**

**# !/bin/bash**

**MY\_KEY=$(cat certificate.pem)**

**echo ${MY\_KEY}**

**echo –e ${MY\_KEY}**

**chmod +x realpem.sh**

**./ realpem.sh**

**Task3: Geçersiz kullanıcıları bulmam gerek. Bunları bir dosyaya atmalıyız**

**cd.. cd task3**

**sadece invalid userdan sonraki kısmı almam gerek**

**Çözüm1: cat auth.log | grep “Invalid user” ya da grep –i “Invalid user” auth.log**

**-**dediğimde tüm invalid userları gördüm benden istediği ınvalid userdan sonraki kısmı kullanıcı isimlerini istiyorum, her bir kullanıcı isminden 3 tane var bu 3 ünden birini çekeceğim ve bazı userlar birkaç kere giriş yapıyor ve kaç kere giriş yaptı bunları nasıl görebilirim ?

**cat auth.log | grep “Failed password for invalid user” | awk ‘ {print $11}’ | sort | uniq –c > result.txt**

-uniq –c ile kimin kaç kere giriş yaptığını saydırdım

**cat auth.log | grep “Failed password for invalid user” | awk ‘ {print $11}’ | sort | uniq | wc –l > invalid\_user.sh**

totalde kaç kere giriş yapılmış

**Çözüm2: grep –Eio “invalid user .+ [preauth]” auth.log | awk ‘ {print $3}’ | sort | uniq –c > invalid\_user2.sh**

**-Bana getireceğin ifade” ınvaled user olsun başında . herhangi bir KARAKTER İFADE demek + birden sonsuza kadar karakter demek sonunda [preauth] ifadesi olsun” mesela \* da sıfırdan sonzuza kadar demek ,? işareti ise istenen karakterden sıfır ve birtane olsun demek REGEX BİLGİSİ ...**

Third\_Part çözümleri

`backup.sh` adlı bir bash script dosyası oluşturalım ve bu script 5 dakikada bir çalışması için cronjob tanımlamalısın, bu script ile `/home/ec2-user/data`, `/etc`, `/boot`, `/usr` dosyaları yedeklenmelidir ve bu veriler `/mnt/backup` klasörüne yerleştirilmesi gerekir. Her dosyayı diğerlerinden ayırman için yedekleme dosyalarına örneğin ana bilgisayar adına ve yedeklemenin tarih-saat-dakikasına göre farklı adlar atamanız gerekir. Her yedekleme dosyası `.tgz` uzantısıyla sıkıştırılmalıdır.

**Çözüm:**

**#!/bin/bash**

# root ayrıcalıklı olup olmadığımızı kontrol edin

**if [[ ${UID} -ne 0 ]]**

**then**

**echo "please run this script as root or with superuser privileges."**

**exit 1**

**fi**

# Hangi dosyaları yedekleyeceğiz. Lütfen /home/ec2-user/data dosyasının mevcut olduğundan emin olun... değişken tanımlayacağım data klasörünın backupını almak için

**DIRECTORIES=(”/home/ec2-user/data”)**

**(**Terminalde ec2 home gel **mkdir data**

**cd data touch file{1..9}.txt** birden dokuza kadar dosya oluşturdu bunların backupını alacağız. backup.sh üçün içinden al home directr. at sonra

**chmod +x backup.sh çalışır duruma getir.)**

# Nereye yedekliyoruz. Lütfen bu scripti çalıştırmadan önce bu dosyayı oluşturun

**BACKUP\_DIR=”/mnt/backup”** (terminalde **cd /mnt/** gelerek **sudo mkdir backup** klsörü oluştur ya da script dosyasının içine bu scripti yazabiliriz **if [[ ! -z ${BACKUP\_DIR} ]]**

(! –d ( böyle bir klasör var mı kontrol et) **then**

**mkdir ${BACKUP\_DIR}**

**fi )**

# Zamana göre arşiv dosya adı oluştur (İNST.HOSTNAME + TARİH)

-Terminalde date –help ile date nasıl formatlayacağını görebilirsin **date +”%Y”** 2023 yılını verdi ay gün saat dakika da eklemek istiyorum **date +”%Y\_%m\_%d\_%H\_%M”** PM şeklinde görmek istiyorsam **date +”%Y\_%m\_%d\_%I\_%M\_%p”** kodunu koşmalıyız. İstersek **date +%F\_%H\_%M** komutunu koşabiliriz (F: FULL DATE DEMEK Y m d)

-Terminalde **hostname** komutunu koştuğumda hostname verecek, belirli bir kısmını almak istiyorum **hostname –s** demeliyim

Bunu script dosyasına uyarlayalım ve backup dir. oluştumu kontrol edelim .

**DATE=$( date +”%Y\_%m\_%d\_%H\_%M”)**

**HOSTNAME= $(hostname –s)**

**ARCHİVE\_FİLE\_NAME=”$HOSTNAME-$DATE.tgz”** (hostname alsın time alsın zip uzantısı)

# Başlangıç ​​durum mesajını yazdır.

echo “archive ptocess is started”

# Dosyaları tar kullanarak yedekleyin. (backup alıyoruz peki neyin backup’nı alacağız)

Terminalde **tar –cvzf** komutunu koşuyorum (-c yeni bir arşiv oluştur isim ver. -z ziple, --gzip , --gunzip --ungzip. -f , arşiv dosyasını kullanır. v verbs. açıklama yap)

Terminalde **sudo tar –cvzf /mnt/backup/sample.gz data** (Data klasörünü tarlayarak Sample.gz olarak mnt backupın altına yerleştirdik )

Bunu script dosyasına uyarlayalım

**tar –cvzf $** **BACKUP\_DIR /$** **ARCHİVE\_FİLE\_NAME $** **DIRECTORIES**

# Bitiş durumu mesajını yazdırın.

**Echo “backup is finished”**

# Dosya boyutlarını kontrol etmek için $dest içindeki dosyaların uzun listesi.

**ls –lh /mnt/backup**

terminalde

**ls ardından ./backup.sh sudo !!**

ardından

**ls /mnt/backup/** çalıştığını gördük

.....................................................................................................................................................

# Bu betiği her 5 dakikada bir çalışacak şekilde ayarlamak için crontab oluşturacağız

**Crontab guru** dan yardım alabiliriz!!

\*\*\*\*\* (her dakika demek)

**\*/5**\*\*\*\* her 5 dakikada bir demek

....................................................................................................................

# /etc/crontab ın içine yazılacak

Terminalde

**ctontab –e**

**\*/5\*\*\*\* sudo /home/ec2-user/backup.sh kaydedip çıkalım**

**ls /mnt/backup/**

**çalıştır...**

.......................

Dosyalarrı sildin nasıl kurtaracağız?

Roota git cd /

ls /mnt/backup/

**Sudo tar –xvf /mnt/backup/ip-172-......... tgz**

(x çıkar demek) dosyalar geri geldi -xvf komutu dosyanın üzerine yazar

Rootta olmam önemli

........................................................

#!/bin/bash

# Define the backup directory

BACKUP\_DIR="/home/ec2-user/backup"

# Define the directories to be backed up

DIRECTORIES=("/home/ec2-user/data")

# Get the hostname of the instance

HOSTNAME=$(hostname)

# Get the current date and time

DATE=$(date +%F-%H-%M)

# Create the backup directory if it doesn't exist

if [ ! -d "$BACKUP\_DIR" ]; then

mkdir "$BACKUP\_DIR"

fi

# Iterate through the directories to be backed up

for DIR in "${DIRECTORIES[@]}" (bütün dosyaların için)

do

# Create the filename for the backup

FILENAME="$BACKUP\_DIR/$HOSTNAME-$DATE-$(basename $DIR).tgz"

# Create the backup using tar

tar -zcvf "$FILENAME" "$DIR"

# Print the backup file name

echo "Backup file: $FILENAME"

done

#!/bin/bash

# Check if we are root privilage or not

if [[ ${UID} -ne 0 ]]

then

echo "Please use this bash script with root privilage"

exit 1

fi

# Which files are we going to back up. Please make sure to exist /home/ec2-user/data file

backup\_files="/home/ec2-user/data /etc /boot /opt"

# Where do we backup to. Please crete this file before execute this script

dest="/mnt/backup"

# Create archive filename based on time

time=$(date +"%Y\_%m\_%d\_%I\_%M\_%p")

hostname=$(hostname -s)

archive\_file="${hostname}-${time}.tgz"

# Print start status message.

echo "We will back up ${backup\_files} to ${dest}/${archive\_file} "

date

echo

# Backup the files using tar.

tar czf ${dest}/${archive\_file} ${backup\_files} # &> /dev/null (Bunu script birkaç kez calistirilip eklenebilir)

# Print end status message.

echo

echo "Congrulations! Your Backup is ready"

date

# Long listing of files in $dest to check file sizes.

ls -lh $dest

-------------

# /etc/crontab ın içine yazılacak

# Good article -> <https://www.cumulations.com/blog/how-to-write-cron-jobs-on-amazon-web-servicesaws-ec2-server/>

# To set this script for executing in every 5 minutes, we'll create cronjob

```bash

crontab -e

```

- vi or nano editor will open. We will run backup.sh script in every 5 minutes. To be able to do this we\'ll write this within opend vi or nano file

```bash

\*/5 \* \* \* \* sudo /home/ec2-user/backup.sh

```

- save and exit from nano or vi

- To check whether your Cron Jobs is saved or not, run the below command.

```bash

$crontab -l

```...............................................................