به نام خدا

مهارت‌های پیشرفته کار با کامپیوتر (بهار ۱۴۰۴)

تمرین کامپیوتری ۳

مهلت ارسال: ۱۴۰۴/۰۳/۱۵

استاد درس: دکتر دوستی

دستیاران طراح: محمدعلی احمدی زارعی - محمدرضا ولی

بازبینی: علی خرم‌فر

قوانین و ملاحظات

نحوه ارسال تمرین:

- تمامی فایل‌ها باید در یک فایل فشرده با نام ECS-CA3-StudentID ارسال شوند.
- کدهای مربوط به هر بخش را با نام مناسب بر اساس جدول انتهای همین فایل ذخیره کرده و همراه گزارش ارسال کنید.
- تمامی کدهای ارسال‌شده باید امکان اجرای مجدد داشته باشند. اگر تنظیمات خاصی برای اجرا نیاز است، آن را ذکر کنید.
- کدهای ارسال‌شده باید توسط خودتان اجرا شده باشند و نتایج اجرا در فایل ارسالی مشخص باشد.

رعایت اصول آکادمیک و صداقت علمی:

- این تمرین می‌تواند به‌صورت **گروهی (دو نفر)** انجام شود، اما ارزیابی آن به‌صورت فردی خواهد بود.
- در صورت مشاهده تشابه در پاسخ، تمامی افراد درگیر نمره صفر دریافت خواهند کرد و موضوع به استاد گزارش خواهد شد.

استفاده از ابزارهای هوش مصنوعی:

استفاده از ابزارهایی مانند ChatGPT ، Gemini، Copilot و موارد مشابه مجاز است، اما تحت شرایط زیر:

- نحوه استفاده از این ابزارها را در گزارش خود توضیح دهید (ابزارهای استفاده‌شده، کاربردهای مشخص و موارد مرتبط).
- تمامی پرامپت‌ها و لینک‌های استفاده‌شده را در انتهای گزارش قرار دهید.
- عدم ارائه این اطلاعات به‌منزله سرقت علمی محسوب شده و منجر به نمره صفر خواهد شد.

مهلت ارسال و جریمه تأخیر:

- امکان ارسال تمرین با تأخیر تا ۲ روز و به ازای هر روز تاخیر ۱۰ درصد جریمه وجود دارد.
- تاخیر به صورت ساعتی محاسبه شده و پس از دو روز تأخیر، تمرین پذیرفته نخواهد شد.

ارزیابی حضوری:

- ارزیابی تمرین به‌صورت حضوری انجام خواهد شد.
- محل ارزیابی: آزمایشگاه NLP ، طبقه منفی یک، دانشکده مهندسی برق و کامپیوتر شماره ۲.

توضیحات تمرین

لطفا پیش از شروع کار بر روی تمرین، به نکات زیر توجه فرمایید.

- حتما ویدئوی راه‌اندازی کلاستر را به دقت مشاهده کنید و مطمئن شوید به کلاستر درس دسترسی دارید.
- آدرس‌های کلاسترهای درس به شرح ذیل است :

dml0: 172.18.32.200

dml1: 172.18.32.201

dml2: 172.18.32.202

dml3: 172.18.32.203

- برای راحتی در توسعه و تست کد، از ماشین مجازی لینوکس خود استفاده نمایید تا ترافیک کلاستر (به خصوص درساعات آخر مهلت تمرین) افزایش نیابد. پس از اطمینان از عملکرد کد، می‌توانید آن را روی کلاستر اجرا کنید.
- در صورت بروز مشکل با ایمیل‌های زیر در ارتباط باشید:

  سوال ۱ و ۲ : m.ahmadizarei@gmail.com

  سوال ۳ : mohammadrezavali78@gmail.com

سوال ۱. داکر – ۳۵ نمره

۱. تولد بی‌بی !

شهین و شاهین میخوان واسه تولد بی‌بی، هدیه‌هایی بهش بدن که دلش شاد بشه.

شهین ازونجایی که میخواد توی مراسم تولد همه بدونن که داره برنامه نویسی یاد میگیره تصمیم گرفته که یه صفحه وب با زبان گولنگ بیاره بالا که توش تولد بی‌بی رو بهش تبریک بگه.

شاهین اما معتقده که بی‌بی خیلی زحمت میکشه و هی بچه اینو تر و خشک کن، واسه اون ناهار درست کن، لباسای اون یکی رو اتو کن و خلاصه همش کار و کار و کار ... به همین خاطر تصمیم گرفته که از روی این سورس کد یک نسخه از بازی ۲۰۴۸ رو دیپلوی کنه تا بی‌بی یه کم بشینه بازی کنه و یه نفسی تازه کنه.

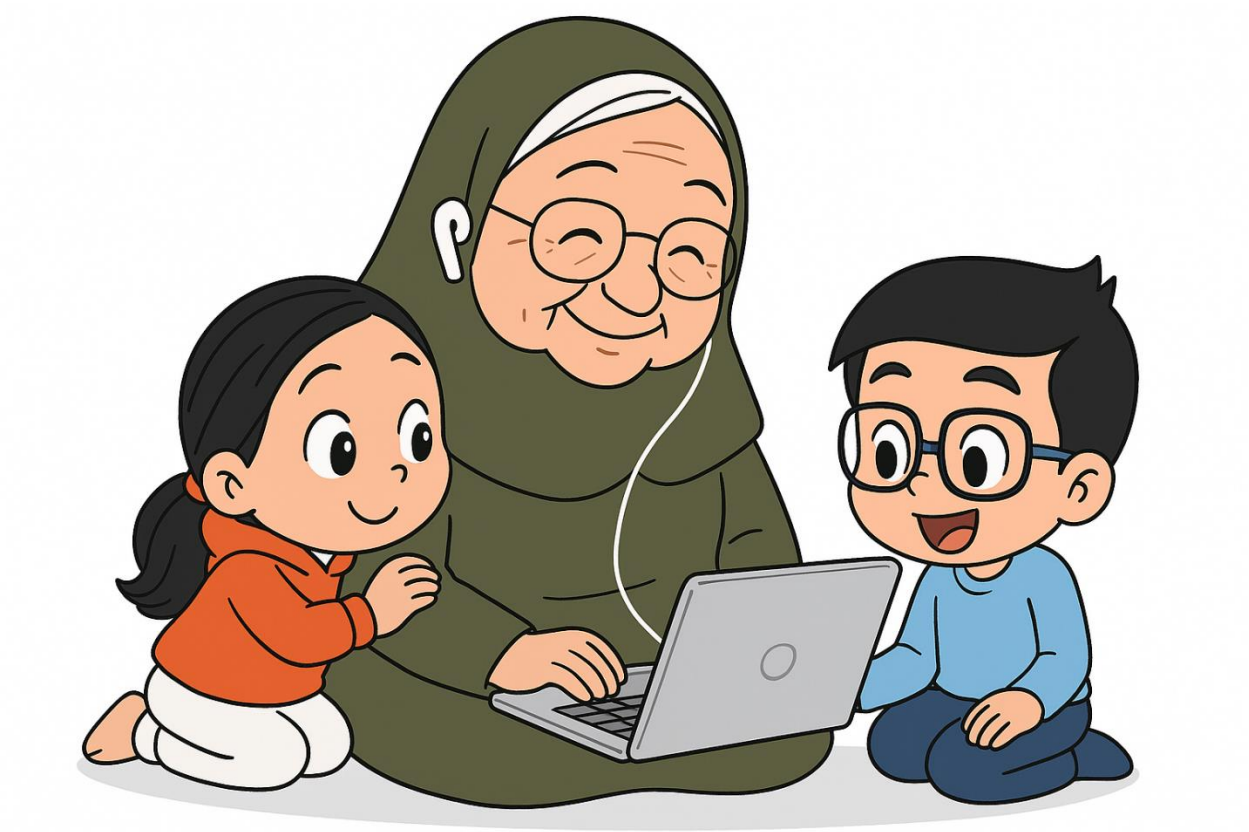
اما مساله ای که هست اینه که بچه‌ها هر دوتاشون میخوان هدیه شون رو روی دامنه Shahin.ir به بی‌بی تقدیم کنند که نمیشه!

واسه حل مشکل رفتن سراغ عمو احمد و اونجا متوجه شدن که میتونن با استفاده از ابزار ریورس پروکسی مثل nginx صفحات بازی شون رو بندازن پشت ریورس پروکسی و روی زیردامنه های مختلف اونها رو بالا بیارن.

بعد از کیک بازی و تمام شدن مراسم تولد بچه‌ها رفتن پیش بی‌بی و ازش خواستن که راستش رو بگه که کدوم هدیه بیشتر خوشحالش کرد. بی‌بی واسه اینکه دل بچه‌ها نشکنه گفت همینکه یادم بودید کلی کیفم رو کوک کرد ولی بچه‌ها خواستن که بی‌بی بگه که با کدومشون بیشتر حال کرده!

بی‌بی هم ازونجایی که جدیدا خودش وارد حوزه tech شده و دستی بر آتش داره و استوری ملت رو رصد میکنه و ... تصمیم گرفت یه چالشی واسه بچه‌ها درست کنه که هم چهارتا چیز از توش یاد بگیرن و هم دست از پرسیدن سوالاتی که تهش شر میشه بردارن.

بی‌بی لپ تاپ رو آورد تا براشون توضیح بده:



بی‌بی پروژه voting-app رو به بچه‌ها معرفی کرد و بهشون توضیح داد که این پروژه یکی از سمپل هایی هست که داکر داره و یه مقدار در مورد معماری میکروسرویس به بچه‌ها توضیح داد.

و نهایتا ازشون خواست که با رعایت نکات زیر این پروژه رو در کنار هدیه‌های خودشون دیپلوی کنند تا یک رای گیری برگزار کنیم و ببینیم هدیه کی جذاب تر بوده!

داکرفایل:

به شاهین و شهین کمک کنید تا برای پروژه هاشون داکرفایل داشته باشند که بتونن کدهایی که دارند رو بیلد کنند و ایمیج داکر مربوط به پروژه خودشون رو داشته باشند.

پروژه شاهین: https://github.com/gabrielecirulli/2048

پروژه شهین هم یه کد گولنگ ساده هست که در ادامه فایل main.go را در اختیار شما میگذاریم:

```go
package main

import (
  "fmt"
  "log"
  "net/http"
)

func handler(w http.ResponseWriter, r *http.Request) {
  fmt.Fprintf(w, "Happy Birthday BIBI JOON!")
}

func main() {
  http.HandleFunc("/", handler)
  log.Println("Starting server on :80")
  err := http.ListenAndServe(":80", nil)
  if err != nil {
    log.Fatal("Error starting server: ", err)
  }
}
```

برای داکرفایل کد شهین سعی کنید تا جای ممکن حجم ایمیج خروجی را کمینه کنید و حجم ایمیجی که میسازید را در فایل تمرین تون گزارش کنید.

داکرکامپوز:

فایل داکر کامپوز پروژه voting-app را تغییر دهید و سه سرویس مربوط به ریورس پروکسی nginx و هدیه شاهین و شهین را به آن اضافه کنید.

به نکات زیر دقت کنید:

- تمامی سرویس‌ها بایستی از روی یک فایل داکرکامپوز واحد دیپلوی شوند.
- اگر وابستگی بین سرویس‌ها وجود دارد بایستی در فایل داکر کامپوز ترتیب بالا آمدن آنها کنترل شود.
- تمامی سرویس‌هایی که لزومی ندارد از بیرون دیده شوند (مثل دیتابیس و ...) بایستی در یک نتورک جداگانه قرار بگیرند.
- تنها پورت ۸۰ از مجموعه سرویس های شما میتواند پابلیش شود و هیچ پورت دیگری نباید به بیرون باز شود.

- با تنظیم dns داخلی لپ تاپ تون زیردامنه های vote و result برای صفحات ووتینگ اپ و همچنین زیردامنه های girl و boy از دامنه Shahin.ir برای صفحات مربوط به کد شهین و شاهین بایستی در ریورس پروکسی کانفیگ شوند.
- تغییر کوچکی در کد voting-app بدهید تا گزینه‌های صفحه رای‌گیری به girl و boy تغییر کنند.
- ازونجایی که بی‌بی دوست نداره بچه‌ها صفحه result رو ببینن که دلگیری بینشون پیش بیاد، در ریورس پروکسی کانفیگی را اضافه کنید که صفحه result برای باز شدن نیاز به یوزر و پسورد داشته باشد.
- یوزر ها را به دلخواه و پسوردها را به صورت حداقل ۳۰ کاراکتر رندم قرار دهید.
- تمامی یوزر و پسورد و متغییرهای محیطی و کانفیگ هایی که در فایل کامپوز هستند بایستی از فایل .env خوانده شوند. زیرا قرار دادن پسورد و موارد مشابه به شکل مستقیم در فایل کامپوز اشتباه است.

نهایتا فایل کامپوز شما با یک دستور docker compose up -d بایستی اجرا شود و هر چهار دامنه‌ای که در بالا توضیح داده شد بر روی مرورگر لپ تاپ شما در دسترس باشند.

۲. سوآرم!

دو ماشین مجازی بر روی لپ تاپ خودتون و دو ماشین مجازی بر روی لپ تاپ هم تیمی تون ستاپ کنید. (برای راحتی میتونید یک بار داکر را نصب کنید و ماشین های دیگه رو از روی اولی کلون بگیرید) با استفاده از این چهار ماشین یک کلاستر سوآرم با سه نود مستر و یک نود ورکر ستاپ کنید. (میتوانید از نتورک bridge بر روی ماشین ها استفاده کنید تا هر چهار ماشین روی شبکه ip بگیرن و همدیگه رو ببینند.)

با استفاده از دستورات swarm وضعیت کلاستری که ایجاد کردید و تعداد نودهای اون رو مشاهده کنید و در گزارش بیاورید.

نهایتا یک اپ ساده مثلا nginx رو روی کلاسترتون دیپلوی کنید و اون رو اسکیل scale کنید به شکلی که پنج نسخه از آن ایجاد شود. (ترجیحا روی هر نود حداقل یک کانتینر موجود باشد.)

امتیازی: (۵ نمره)

در مورد نحوه دسترسی به این اپلیکیشن از بیرون توضیح دهید. درخواست ها به چه شکل سرویس دهی می‌شوند. آدرس ip کدام ماشین را بایستی در dns سرور قرار دهیم.

سوالات تشریحی (امتیازی: ۱۰ نمره)

۱) میدانیم موقع توسعه کد اگر فرآیند بیلد را با داکر انجام دهیم، بعد از هر تغییر بایستی مجدد docker build بزنیم تا تغییرات کدمان را زمان اجرا ببینیم. حالا داکر در کنفرانس dockercon 2023 قابلیت جدیدی را تحت عنوان docker compose watch معرفی کرده. در مورد این قابلیت توضیح دهید برای اجرای عملی میتوانید از پروژه آواتار که در

سمپل های داکر وجود دارد استفاده کنید. برای گزارش این قسمت میتونید پس از اجرای دستور docker compose watch در این پروژه قسمتی از کد (مثلا رنگ یکی از بخش‌ها در فایل css) را تغییر دهید و نشان دهید که تغییرات به صورت مستقیم بر روی پروژه اعمال می‌شوند.

۲) در مورد مفاهیم docker bake و wasm docker تحقیق کنید و موارد استفاده هرکدام را توضیح دهید. اگه مثال عملی بزنید که دیگه عالی!

۳) در مورد تفاوت RUN و COMMAND ، WORKSPACE و ENTRYPOINT ، COPY و ADD در داکرفایل توضیح دهید. همچنین بیان کنید که کدام یک از اجزای داکرفایل منجر به اضافه شدن یک لایه موثر و حجم دار به ایمیج داکر در خروجی می‌شوند.

سوال ۲. کوبرنتیز – ۳۵ نمره

کوبرنتیز ابزاری که واسه اوپن سورس کردنش یه کمپانی زدن!

اگه تا حالا از تاکسی های اینترنتی استفاده کردید یا از فروشگاه های آنلاین بزرگ چیزی خرید کردید یا موقع وب گردی توی تبلیغات سایت ها همون کالایی که اخیرا سرچش کردید رو دیدین، احتمالا شما به عنوان کاربر یکی از شرکت های بزرگ حوزه IT ایران در حال دریافت سرویس از یک کلاستر کوبرنتیز بودید!!!

توصیه میکنم این دوتا ویدیو رو که نشون میدن چطوری این ابزار قدرتمند شکل گرفت و چه ماموریت مهمی رو داشت حتما ببینید. (از ویدیوها سوال نمیاد و به تمرین کمکی نمیکنن فقط واسه درک بهتر خودتون و آشنا شدن با فضای داستانه.)

The official Kubernetes Documentary part1

The official Kubernetes Documentary part2

شرکت محترم هم‌روش هم که یکی از استارتاپ‌های در حال رشد قوی ایرانی در حوزه کلاد هست این ویدیوهارو زیر نویس کرده که اینجا (قسمت اول و قسمت دوم) میتونید ببینید.

نکته بسیار مهم:

با توجه به فرصت محدود این درس و زمان اندکی که ممکن بود برای این ابزارها اختصاص داده بشه و همچنین با توجه به حجم زیاد مطالب و نکات ریزی که در داکر و مخصوصا کوبرنتیز داریم، حتما لازم میدونم که اینجا یادآوری کنم که اگر تمایل به یادگیری این ابزارها دارید، لازم هست تا زمان بیشتری رو صرف کنید تا پوشش بهتری رو روی مطالب اونها داشته باشید. نهایتا در این تمرین تلاش میکنم تا با گوشه‌ای از قابلیت‌های این ابزار آشنا بشیم.

خب دیگه دست به کیبورد شیم!

خب برای اینکه با کوبرنتیز کار کنیم، نیاز به یه کلاستر کوبرنتیز داریم، اما ازونجایی که راه‌اندازی یک کلاستر کوبرنتیز با تمام ویژگی نیازمند منابع و تخصص هست و اندکی پیچیدگی داره. برای شروع و تمرین اولیه کوبرنتیز میریم به سراغ ابزارهایی که به ما یک کلاستر سینگل نود میدن! که میشه از اون برای تست کوبرنتیز و یا برای موارد آموزشی استفاده کرد.

توی این تمرین هم ما از یکی از ابزارهای سینگل نود کوبرنتیز استفاده می‌کنیم.

(توصیه بنده ابزار kind (Kubernetes in Docker) هست، اما شما میتونید از ابزارهای دیگه مثل k3s ، k3d ، minikube و microk8s استفاده کنید.)

بنابراین با استفاده از یکی ازین ابزارها تمرین تون رو پیش ببرید. برای نصب kind میتونید از دستورات زیر در ترمینال استفاده کنید.

```
# For AMD64 / x86_64
[ $(uname -m) = x86_64 ] && curl -Lo ./kind
https://kind.sigs.k8s.io/dl/v0.29.0/kind-linux-amd64chmod +x ./kind
sudo cp ./kind /usr/local/bin/kind
rm -rf kind
```

با دستورات زیر میتونید چک کنید که نصب تون انجام شده باشه.

```
which kind
kind --version
```

حال با استفاده توضیحات مربوط به کانفیگ kind فایل کانفیگی برای ایجاد یک کلاستر با سه نود مستر و یک نود ورکر ایجاد کنید و کلاستر رو راه‌اندازی کنید.

- توضیح دهید که چرا تعداد نودهای master رو عدد فرد انتخاب می‌کنند؟ همچنین در مورد تعداد نودهای مستر و ورکر و ... در کلاسترهای بزرگ در سطح Enterprise تحقیق کنید.

آپدیت ما نباید مشتری رو اذیت کنه!

در مورد تفاوت Replication Controller و Replica Set تحقیق کنید.

با توجه به درکی که از Replication Controller بدست آوردید، حدس میزنید اگر دوتا ریسورس از این جنس در کلاستر کوبرنتیز ایجاد کنیم که کاملا یکسان باشند و تنها نام متفاوتی داشته باشند. در حالیکه قسمت replicas اونها برابر ۱ هست، در

نهایت چند پاد در کلاستر باقی میماند؟ (برای درک بهتر این قسمت در ادامه یک مانیفست نمونه براتون قرار میدهیم، اما مهم هست که قبل از apply کردن اون بر روی کلاسترتون به این سوال فکر کنید و انتظاری که دارید رو بدونید.)

```yaml
apiVersion: v1
kind: ReplicationController
metadata:
  name: nginx-rc-one
spec:
  replicas: 1
  selector:
    app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx:alpine
        ports:
        - containerPort: 80
---
apiVersion: v1
kind: ReplicationController
metadata:
  name: nginx-rc-two
spec:
  replicas: 1
  selector:
    app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx:alpine
        ports:
        - containerPort: 80
```

حال فرض کنید به طور مشابه اپلیکیشن شما با یکی از همین ریسورس‌ها (Replication Controller یا Replica Set) روی کلاستر کوبرنتیز اومده بالا (مثلا میتونید از یک نسخه nginx یا هر برنامه دلخواه دیگه استفاده کنید.) و در یک نسخه در حال

حاضر قرار دارد مثلا nginx:1.20 و شما در حال توسعه اپلیکیشن تون هستید و زمان آن رسیده تا نسخه بعدی آن را به مشتری نشان دهید! یعنی مثلا nginx:1.21 رو جایگزین قبلی کنید بر روی کلاسترتون، چه ایده‌هایی دارید؟ چه تغییراتی در مانیفست تون ایجاد میکنید و به چه شکل این آپدیت رو انجام میدید؟ (فاکتورهایی مثل نداشتن down time به این معنا که سرویس همواره در دسترس باشد و میزان مصرف منابع جزو مواردی هستند که باید بهشون دقت کنیم)

حال به سراغ ریسورس دیگری در کوبرنتیز می‌رویم.

در مورد Deployment تحقیق کنید و نحوه کارکرد اون و تفاوتی که با Replica Set دارد توضیح دهید. برای اپلیکیشن خودتون یک مانیفست deployment بنویسید. و مجددا سناریو آپدیت رو سعی کنید پیاده سازی کنید.

در مورد انواع استراتژی های roullout در deploymenet ها تحقیق کنید و تفاوت اونها رو توضیح دهید. دو استراتژی recreate و rolling update را روی مانیفستی که برای اپلیکیشن تون نوشتید تست کنید و نحوه عملکرد اون رو در عمل نشان دهید.

کمپین فروش، جمعه سیاه، شب عید و ...

نیازمندی در شرکت‌ها به وجود میاد که در زمان‌های خاصی ممکنه که تا چندین برابر حالت عادی نیاز باشه که درخواست‌های کاربرانشون رو پاسخ بدن. افزایش منابع و آمادگی همیشگی برای پاسخگویی هزینه بر هست، از طرفی عدم پاسخگویی به چنین نیازی معنیش ضرر مستقیم به بیزینس هست.

حالا در کنار این نیازمندی با توسعه سیستم‌های مبتنی بر ابر که امکانی تحت عنوان pay as you go رو بهمون میدن، جای ابزاری که بتونه ازین قابلیت به خوبی استفاده کنه و به وقت نیاز ساختار رو بزرگ کنه تا پاسخگوی مشتری باشه خالیه.

کوبرنتیز با استفاده از قابلیت اسکیل کردن خودکار یا Auto Scaling (HPA و VPA) این کار رو انجام میده. در مورد این ریسورس ها در کوبرنتیز بخونید.

در این تمرین یک deployment از یک ایمیج دلخواه (مثلا همون nginx) با پنج رپلیکا ایجاد کنید. با انتخاب نوع مناسبی از service در کوبرنتیز دسترسی نتورکی به پادهایی deployment تون ایجاد کنید. سپس سازوکار مناسب برای افزایش خودکار پادها تا تعداد ۱۰ و کاهش اونها تا تعداد ۳ رو ایجاد کنید. مثلا اگر میزان متوسط مصرف cpu پادها از ۷۰ درصد بالاتر رفت یا اگر میزان مصرف مموری پادها از ۸۰ درصد بالاتر رفت آنگاه کوبرنتیز تشخیص دهد که باید تغییری در تعداد اونها ایجاد کنه. با استفاده از ابزاری مثل ab (Apache Benchmark) یک لود ساختگی روی پادهاتون بندازید و درستی کارکرد فرآیند Autoscaling را بررسی کنید و نشان دهید. (دقت کنید که میتونید به کمک limit range میزان منابع پادهاتون رو محدود کنید تا راحت تر تست رو انجام بدید. همچنین دقت کنید که ریسورس های اسکیل به صورت خودکار نیازمند اضافه کردن قابلیتی به کلاستر هستند که از طریق اون بتوانند متریک های لازم برای بررسی وضعیت مصرف منابع پادها رو داشته باشند بنابراین لازم هست که این قابلیت رو به کلاسترتون اضافه کنید.)

سوالات تشریحی (امتیازی: ۱۰ نمره)

۱) در مورد PV و PVC تحقیق کنید و توضیح دهید که StorageClass چه کاربردی در کلاستر کوبرنتیز دارد؟

۲) درمورد انواع CNI تحقیق کنید و تفاوت بین Cilium و Calico را به عنوان دو cni معروف بیان کنید.

۳) با استفاده از affinity ساختاری ایجاد کنید که بر روی هرکدوم از نودهای کلاستری که دارید یک پاد redis در کنار یک پاد nginx قرار بگیرد و این مورد رو در عمل تست کنید.

۴) در مورد KEDA تحقیق کنید و توضیح دهید که چه کاربردی دارد؟

سوال ۳. PORT SCANNING – ۳۰ نمره

هدف از پایش پورت ها شناسایی پورت های باز، بسته و یا فیلتر شده سیستم‌ها است. این مرحله به مهاجمان کمک می‌کند تا نقاط ضعف و نقاطی که پتانسیل نفوذ به سیستم ها را دارند شناسایی شوند و برای حملات احتمالی استفاده شوند. بر همین اساس برای دفاع و حفاظت در برابر حملات، پایش پورت‌ها به مدیران شبکه کمک می‌کند تا از آسیب‌پذیری‌های احتمالی سیستم‌ها مطلع شده و برای آن چاره‌اندیشی کنند. به طور مثال کلیک بر یک فایل ضمیمه ایمیل آلوده می‌تواند منجر به نصب یک برنامه سرور روی سیستم شود و امکان دسترسی به مهاجمان دهد. پایش پورت‌ها و شناسایی پورت‌های باز امکان شناسایی برنامه‌های در حال اجرا روی پورت‌ها و بررسی آسیب پذیری‌های احتمالی آن‌ها را فراهم می‌کند.

در این بخش قصد داریم یک برنامه با استفاده از زبان Python و کتابخانه scapy یا dpkt بنویسیم که یک فایل PCAP را به منظور شناسایی حملات احتمالی SYN scan تحلیل می‌کند. به این منظور یک فایل نمونه PCAP در اختیار شما قرار گرفته است که می‌توانید با استفاده از Wireshark نیز محتوای آن را بررسی کنید. (توجه داشته باشید که فایل مذکور شامل تعداد زیادی بسته می‌شود و ممکن است هنگام پردازش آن توسط سیستم خودتان با مشکل محدودیت منابع مواجه شوید؛ به این دلیل می‌توانید از نسخه به شدت ساده شده آن (صرفا برای تست کد خود در مراحل اولیه) استفاده کنید و در صورت مشکل با PCAP اصلی می‌توانید با استفاده از Wireshark فایل خود را Generate کرده و صحت کد خود را با استفاده از آن بررسی کنید؛ در صورت انجام این تمرین با PCAP تولید شده توسط خودتان، حتما این فایل را به عنوان بخشی از پاسخ خود در ایلرن آپلود کنید)

۱. شرح مسئله

Network trace ضبط ترافیک شبکه در یک بازه زمانی مشخص است که شامل بسته‌های ارسالی و دریافتی بین سیستم‌ها می‌شود و نتیجه آن به صورت فایل PCAP قابل دسترسی می‌باشد که می‌تواند شامل انواع بسته‌های شبکه مانند Ethernet, IP و TCP باشد.

در یکی از تکنیک‌های port scan، که به عنوان SYN scan شناخته می‌شود، scanner بسته‌های TCP SYN (اولین بسته در TCP handshake) را ارسال کرده و به دنبال پاسخ‌های SYN+ACK (دومین مرحله‌ی TCP handshake) از سمت میزبان می‌گردد. از آنجا که اکثر میزبان‌ها آماده دریافت ارتباط در تمام Portها نیستند، در یک SYN scan تعداد بسته‌های SYN+ACK دریافتی معمولاً کمتر از بسته‌های SYN ارسال‌شده است. با مشاهده این تفاوت در یک Network trace می‌توان مبدا پکت‌هایی را که احتمالا در حال انجام port scanning هستند، شناسایی کرد.

۱. تحلیل ترافیک

برنامه باید فایل PCAP که شامل بسته‌های ضبط‌شده شبکه است را بخواند و تحلیل کند. تنها بسته‌های معتبر شامل Ethernet, IPو TCP مورد بررسی قرار گیرند. همچنین بسته‌های Malformed یا بسته‌هایی که از این پروتکل‌ها استفاده نمی‌کنند، باید نادیده گرفته شوند.

۲. شناسایی IPهای مورد نظر

برنامه باید تعداد بسته‌های SYN ارسال شده توسط هر منبع (IP address) و تعداد بسته‌های SYN+ACK دریافتی از هدف را محاسبه کند. همچنین در صورتی که یک IP به‌طور مکرر (بیش از ۳ برابر تعداد SYN+ACK دریافتی) بسته‌های SYN ارسال کرده باشد، آن IP باید به‌عنوان منبع احتمالی SYN scan شناسایی شود.

۳. ورودی و خروجی

برنامه شما باید یک آرگومان ورودی داشته باشد (نام فایل PCAP برای تحلیل) مطابق زیر:

```
python syn_scanner.py sample.pcap
```

خروجی نیز باید مجموعه‌ای از IP addressهای مشکوک به صورت زیر طبق توضیحات b.۲ باشد. (ترتیب IPها ممکن است با هر بار اجرا متفاوت باشد)

۱۲۸,۳,۲۳,۲
۱۲۸,۳,۲۳,۵
۱۲۸,۳,۲۳,۱۱۷
۱۲۸,۳,۲۳,۱۵۸
۱۲۸,۳,۱۶۴,۲۴۸
۱۲۸,۳,۱۶۴,۲۴۹

صحت کد شما با استفاده از چند نمونه فایل PCAP مورد ارزیابی قرار خواهد گرفت.

۱. سوالات تشریحی (امتیازی: ۵ نمره)

۱) فایل PCAP اصلی را به صورت دستی با استفاده از Wireshark تحلیل کرده و بیان کنید آیا اطلاعات دیگری در Trace مربوطه وجود دارد که شخص مهاجم بتواند به نحوی از آن استفاده کند؟ درباره راه‌های استفاده از آن توسط شخص مهاجم و آسیب‌پذیری‌های احتمالی توضیح دهید؛ همچنین سناریوی حمله با استفاده از این آسیب‌پذیری‌ها را در گزارش خود تحلیل کنید. (توجه داشته باشید این سوال می‌تواند جواب‌های بسیار متفاوتی داشته باشد و نکته حائز اهمیت درک و تحلیل شما از Trace یک سناریوی تبادل اطلاعات در شبکه است)

نحوه تحویل تمرین کامپیوتری ۳

فایلها را به صورت زیر نام‌گذاری کرده و همه را در یک فایل zip در سامانه ارسال کنید.

سوال	بخش	نام فایل‌ها
۱	**تمام بخش‌ها**	تمامی موارد مربوط به این قسمت را در یک پوشه با نام P1 قرار دهید: فایل های پروژه شامل: داکرفایل ها داکر کامپوز فایل کانفیگ و ... گزارش قدم به قدم انجام تمرین به همراه Screenshotها، شرح دستورات و پاسخ سوالات تشریحی در قالب یک فایل PDF
۲	**تمام بخش‌ها**	تمامی موارد مربوط به این قسمت را در یک پوشه با نام P2 قرار دهید: گزارش قدم به قدم انجام تمرین به همراه Screenshotها، شرح دستورات و پاسخ سوالات تشریحی در قالب یک فایل PDF
۳	**تمام بخش‌ها**	تمامی موارد مربوط به این قسمت را در یک پوشه با نام p3 قرار دهید: - فایل برنامه - فایل PCAP (در صورت لزوم) - گزارش قدم به قدم انجام تمرین به همراه Screenshotها، شرح دستورات، پاسخ سوالات تشریحی و باگ‌ها (در صورت وجود) در قالب یک فایل PDF