

به نام خدا

مبانی رایانش ابری (نیمسال دوم تحصیلی ۹۹-۱۴۰۰)

تمرین شماره ۱: پیاده‌سازی سرور مدیریت ماشین مجازی مبتنی بر VirtualBox

آخرین تاریخ آپلود پاسخ در **courses**:

ساعت ۲۳:۵۹، ۲۰ فروردین ۱۴۰۰

بخش ۱

- الف) مجازی سازی را تعریف کنید و سپس نقش آن را در گسترش رایانش ابری بیان کنید.
- ب) درباره چنداجاره‌ای^۱ در رایانش ابری تحقیق کنید و آن را مختصراً شرح دهید. مشکلات موجود در این روش را نام برده و توضیح دهید.
- ج) درباره مجازی‌سازی سخت افزاری^۲ تحقیق کنید و انواع آن را با ذکر مثال تشریح کنید.

بخش ۲

هدف از این بخش، پیاده سازی یک سرور مدیریت ابر به صورت Restful API است. این سرور امکانات مختلفی را فراهم می‌کند که در ادامه بیان می‌شوند.

تمامی مراحل زیر را در کامپیوتر میزبان خود انجام دهید و برای هر مرحله تصاویر (screenshot) مناسب را در گزارش خود بگنجانید.

۱-۲ ایجاد ماشین مجازی (تنها بخش کوچکی از بارم این تمرین)

۱. VirtualBox را بر روی سیستم میزبان خود نصب کنید.
۲. یک ماشین مجازی Ubuntu Server را بر روی سیستم میزبان ایجاد کنید. دقت کنید حتماً نام آن را VM1 بگذارید. تنظیمات این ماشین مجازی را به طوری قرار دهید که برای نصب و اجرای آن مشکلی پیش نیاید.
۳. پیکربندی شبکه ماشین مجازی را طوری تنظیم کنید که VM1 از سیستم میزبان قابل دسترسی باشد. و همچنان از داخل VM1 بتوانید به اینترنت دسترسی داشته باشید. سرویس SSH را در VM1 اجرا کنید تا بتوانید از میزبان به آن SSH بکنید و این را در گزارش خود نشان دهید.
۴. ماشین مجازی دیگری همانند VM1 با نام VM2 ایجاد کنید. دقت کنید که این بخش تنها وقتی قابل پذیرش است که VM2 از روی VM1 ساخته شده باشد (در مورد دستور clone مطالعه کنید).
۵. پیکربندی شبکه و مراحل برقراری ارتباط SSH با VM2 را نیز انجام دهید.

۲-۲ سرور مدیریت ماشین مجازی

۱. سرور شما باید از پروتکل HTTP استفاده کند.
۲. پس از انتخاب زبان برنامه نویسی مورد نظر، کتابخانه‌ای برای ارتباط به VirtualBox مربوط به همان زبان پیدا کنید.

¹ Multitenancy

² Hardware-level virtualization

۳. تمامی درخواست های ارسالی به یک مسیر(endpoint) ارسال شوند.(مثلا: <http://localhost:8000/>)
۴. سرور شما باید یک سیستم احراز هویت ساده داشته باشد. به این معنی که برای دسترسی به امکانات سرور، کاربر باید ابتدا وارد سیستم شده و در درخواست های بعدی خود، توکن احراز هویت را به سرور ارسال کند. در احراز هویت، نام کاربری و رمز را دریافت و یک توکن خاص برای آن کاربر ایجاد کنید. سپس توکن را در سرور خود به عنوان شاخص آن کاربر ذخیره کنید و در آخر توکن را برای کاربر بازگردانید. در درخواست های بعدی توکن را باید در قسمت header درخواست خود قرار دهید و در سمت سرور با استفاده از توکن تشخیص دهید که کاربر درخواست کننده چه کسی است (یا اینکه چه دسترسی هایی دارد).
۵. سرور شما باید یک سطح دسترسی ساده نیز تعریف کند. سیستم یک کاربر admin دارد که می تواند تمام دستورات را به صورت نامحدود اجرا کند و همچنین یک کاربر دیگر با نام user1 دارد که تنها می تواند دستورات را روی ماشین مجازی VM1 اجرا کند.
۶. برای آزمایش سرور خود میتوانید از نرم افزار Postman استفاده کنید. این نرم افزار قابلیت های لازم جهت ارسال درخواست های HTTP در اختیار شما قرار میدهد. برای گرفتن نمره مرتبط با هر دستور، بایستی که بخش تست آن را نیز پیاده کنید و نتیجه را در گزارش خود قرار بدید و همچنین اماده ارائه اسکایپی باشید.
۷. حال سرور را طوری تنظیم کنید که بتواند پاسخگوی موارد زیر باشد:
- a. وضعیت(خاموش، روشن، ...) همه ماشین های مجازی
 - b. وضعیت یک ماشین مجازی خاص
 - c. امکان روشن و خاموش کردن یک ماشین مجازی
 - d. تغییر تعداد هسته های پردازنده و میزان حافظه اصلی یک ماشین مجازی
 - e. ایجاد یک ماشین مجازی با استفاده از قابلیت clone کردن
 - f. حذف یک ماشین مجازی
 - g. ارسال یک فرمان به ماشین مجازی و بازگرداندن نتیجه آن
 - h. بارگذاری فایل روی ماشین مجازی
 - i. انتقال فایل از یک ماشین مجازی به یک ماشین مجازی دیگر
- در ادامه توضیح مختصری درباره هر بخش داده شده است.
- گرفتن وضعیت:** سرور شما باید بتواند با گرفتن نام یک ماشین مجازی، وضعیت فعلی آن ماشین را برگرداند. برای مثال، سرور شما با گرفتن json زیر، باید درخواست را متوجه شود و وضعیت فعلی ماشین مجازی VM1 را بازگرداند. فرمت خروجی را نیز مشاهده می کنید. وضعیت ماشین های مجازی را به صورت زیر بازگردانید:

روشن: on

خاموش: off

در حال روشن شدن: powering on

در حال خاموش شدن: powering off

ورودی:

```
{
  "command": "status",
  "vmName": "VM1"
}
```

خروجی:

```
{
  "command": "status",
  "vmName": "VM1",
  "status": "On"
}
```

برای گرفتن تمام وضعیت ها از فرمت ورودی و خروجی زیر پیروی کنید:

ورودی:

```
{
  "command": "status"
}
```

خروجی:

```
{
  "command": "status",
  "details": [
    {
      "vmName": "VM1",
      "status": "On"
    },
    ...
  ]
}
```

روشن و خاموش کردن ماشین مجازی: سرور شما باید بتواند با گرفتن نام یک ماشین مجازی آن را خاموش یا روشن کند.

ورودی:

```
{
  "command": "on/off",
  "vmName": "VM1"
}
```

خروجی:

```
{
  "command": "on/off",
  "vmName": "VM1",
  "status": "powering on/powering off"
}
```

تغییر تنظیمات منابع ماشین مجازی: سرور شما باید بتواند تنظیمات مربوط به پردازنده و حافظه اصلی اختصاص داده شده به یک ماشین مجازی را تغییر دهد.

ورودی:

```
{
  "command": "setting",
  "vmName": "VM1",
  "cpu": 2,
  "ram": 1024
}
```

خروجی:

```
{
  "command": "setting",
  "vmName": "VM1",
  "cpu": 2,
  "ram": 1024,
  "status": "Ok"
}
```

ایجاد یک ماشین مجازی: سرور شما باید بتواند از یک ماشین مجازی موجود، یک ماشین مجازی دیگر بسازد.(clone)

ورودی:

```
{
  "command": "clone",
  "sourceVmName": "VM1",
  "destVmName": "VM2"
}
```

خروجی:

```
{
  "command": "clone",
  "sourceVmName": "VM1",
  "destVmName": "VM2",
  "status": "Ok"
}
```

حذف یک ماشین مجازی: سرور شما باید بتواند با گرفتن نام یک ماشین مجازی آن را حذف کند.

ورودی:

```
{
  "command": "delete",
  "vmName": "VM1"
}
```

خروجی:

```
{
  "command": "delete",
  "vmName": "VM1",
  "status": "Ok"
}
```

ارسال فرمان به ماشین مجازی: سرور شما باید بتواند یک فرمان قابل قبول برای terminal دریافت کرده، آن را در ماشین مجازی مورد نظر اجرا کند و نتیجه را بازگرداند.

ورودی:

```
{
  "command": "execute",
  "vmName": "VM1",
  "input": "mkdir sina && touch sina.txt && ls"
}
```

خروجی:

```
{
  "command": "execute",
  "vmName": "VM1",
  "status": "Ok",
  "response": "sina sina.txt"
}
```

بارگذاری فایل روی یک ماشین مجازی: سرور شما باید بتواند یک فایل، مسیر و ماشین مجازی مد نظر را به عنوان ورودی گرفته و فایل را در مسیر گفته شده در ماشین مجازی خواسته شده قرار دهد. دقت کنید که نوع ورودی فایل و دیگر ورودی ها متفاوت است که شما باید این مورد را در نظر بگیرید.

انتقال فایل از یک ماشین مجازی به یک ماشین دیگر: سرور شما باید مسیر یک فایل در ماشین مجازی اول را به مسیر خواسته شده در ماشین مجازی دوم انتقال دهد.

ورودی:

```
{
  "command": "transfer",
  "originVM": "VM1",
  "originPath": "/home/sina.txt",
  "destVM": "VM2",
  "destPath": "/home/temp/"
}
```

خروجی:

```
{
  "command": "transfer",
  "originVM": "VM1",
  "originPath": "/home/sina.txt",
  "destVM": "VM2",
  "destPath": "/home/temp/",
}
```

نحوه تحویل تمرین ۱

موارد زیر را در قالب یک فایل زیپ با نام «HW1_student_id.zip» در صفحه درس بارگذاری کند.

۱. گزارشی که باید شامل پاسخ به بخش اول، توصیف نکات طراحی و پیاده‌سازی، و توضیحات شما در مورد نحوه انجام مراحل بیان شده باشد. گزارش شما بایستی که از کیفیت خوب برخوردار بوده و از تکرار یا بی‌نظمی پرهیز کنید.

۲. کدهای استفاده شده در پیاده‌سازی. (تمیز و خوانا بودن کدها از اهمیت بالایی برخوردار است)

- این تمرین تحویل اسکایی دارد.

- این تمرین فردی است.

جریمه دیرکرد

هر روز تاخیر در ارسال تمرین ۱۰٪ نمره منفی خواهد داشت. امکان بارگذاری تمرین تنها تا ۵ روز بعد از تاریخ تعیین شده ممکن خواهد بود.

جریمه تقلب

۱. همه بایستی که خود تمرین را انجام دهند و هرگونه تقلب یا ارسال کار دیگران یا کارهای موجود در وب که بخش برنامه‌نویسی تمرین را به شکل جزئی یا کلی انجام داده است، غیرقابل پذیرش و عواقب شدیدی خواهد داشت. دانشجویان بی‌شک می‌توانند از راهنمای موجود در وب یا کتابخانه‌های کمکی استفاده نکنند تا جایی که همه منابع و کتابخانه‌ها کمکی به صراحت ذکر شده باشد.

۲. بنده و گروه حل تمرین تمام تلاش خود را برای شناسایی تقلب‌های احتمالی خواهیم کرد تا در نهایت یک ارزیابی عادلانه از همه دانشجویان عزیز داشته باشیم. ما از MOSS برای شناسایی فایل‌های مشابه استفاده خواهیم کرد.

۳. در صورت شناسایی تقلبی که ۵۰٪ یا پایین‌تر از کار را شامل می‌شود، اعضای گروه اخطار نوع یک دریافت کرده و نمره «۰.۵- *» بارم تمرین ۱» به اعضای گروه تعلق می‌گیرد و در صورت شناسایی تقلبی که بیشتر از ۵۰٪ کار را پوشش می‌دهد به اعضای گروه اخطار نوع دوم تعلق گرفته و نمره «۱- *» بارم تمرین ۱» به اعضای گروه تعلق می‌گیرد. علاوه بر این نمره منفی، گرفتن چند اخطار، عواقب شدیدتری خواهد داشت.

در نهایت، هرگونه سوال در مورد تمرین و بخش‌های آنها را تنها و تنها از طریق سایت درس و ایجاد مباحثه با عناوین مرتبط مطرح بفرمایید.

تندرست و موفق باشید

تیم درس مبانی رایانش ابری