

دانشگاه صنعتی اصفهان

مرکز ابررایانش

## گزارش کارآموزی



**مرکز ابر رایانش ملی شیخ بهایی**

National High Performance Computing Center



نویسنده: مجتبی ملائی

استادکارآموزی: دکتر فاطمه دلدار  
سرپرست کارآموزی: آقای محمد صالح محمدی

۲۸ شهریور ۱۴۰۴

## چکیده

من و همکارانم در این دوره کارآموزی ضمن آشنایی با مرکز ابررایانش و افراد متخصص شاغل در این مرکز با استفاده از ابزار های kubernetes, kubervirt و ceph بستری ایجاد کردیم تا بتوانیم ماشین های مجازی ویندوز را با هدف در دسترس بودن کامل<sup>۱</sup> و با قابلیت هایی مانند تخصیص منابع به صورت برخط و امکان خودکار سازی<sup>۲</sup> به صورت ایمن در اختیار مصرف کنندگان قرار دهیم.

# فهرست مطالب

۱	معرفی	۲
۱.۱	پیش زمینه من	۲
۲.۱	مرکز ابررایانش شیخ بهائی	۲
۱.۲.۱	منظور از یک خوشه HPC چیست؟	۲
۲.۲.۱	معماری	۳
۳.۲.۱	نحوه کار	۳
۳.۱	پروژه	۴
۲	آماده سازی	۵
۱.۲	آماده سازی کامپیوترها	۵
۲.۲	آماده سازی سرورها	۵
۳	ارکستراسیون	۶
۱.۳	Kubernetes	۶
۴	پیاده سازی شبکه	۸
۱.۴	calico	۸
۲.۴	multus	۸
۳.۴	macvtap	۸
۴.۴	bridge	۸
۵.۴	kubemacpool	۸
۵	بستر مجازی سازی در k8s	۹
۱.۵	Kubevirt	۹
۲.۵	منابع ماشین ها	۹
۳.۵	انتقال زنده	۹
۴.۵	تخصیص منابع به صورت زنده	۹
۶	Windows image	۱۰
۱.۶	گرفتن اطلاعات	۱۰
۲.۶	تنظیمات اضافی	۱۰
۷	kubevirt-manager	۱۱
۸	نظارت	۱۲
۱.۸	Prometheus	۱۲
۲.۸	Graphana	۱۲
آ	اطلاعات اضافی	۱۳

# فصل ۱

## معرفی

### ۱.۱ پیش زمینه من

من کارآموزی‌ام را در بعد از ترم ششم آغاز کردم. در این دوران با برخی از گرایش‌ها، زمینه‌ها و مسیرهای مختلف در حوزه کامپیوتر آشنا شدم که از جمله آنها برنامه نویسی سمت سرور<sup>۱</sup>، امنیت سایبری، شبکه و هوش مصنوعی بود. در این میان به فناوری‌های سمت سرور مانند Linux علاقه‌مند شدم و این موضوع باعث شد تا تصمیم بگیرم که کارآموزی‌ام را در حوزه‌های مربوط به سرورها بگذرانم.

### ۲.۱ مرکز ابررایانش شیخ بهائی

شبکه ملی محاسبات ابری در سال ۱۳۸۸ با هدف ارائه خدمات کامل و جامع محاسباتی به صنایع و مراکز علمی و تحقیقاتی کشور و با حمایت معاونت علمی و فناوری ریاست جمهوری افتتاح شد. مرکز ملی محاسبات ابری<sup>۲</sup> (NHPCC) به عنوان یکی از چندین مرکز ابررایانه کشور، در آن زمان بخشی از این طرح ملی را در قالب تأسیس مرکز ملی ابررایانه شیخ بهائی اجرا کرد. تمهیدات در نظر گرفته شده در مرکز ملی محاسبات ابری شیخ بهائی به گونه‌ای است که امکانات سخت‌افزاری و نرم‌افزاری جامع و کامل و همچنین خدمات آموزشی و مشاوره‌ای فنی و تخصصی برای استفاده بهینه (در محل یا از راه دور) در دسترس باشد.

### ۱.۲.۱ منظور از یک خوشه HPC چیست؟

محاسبات با کارایی بالا یا HPC<sup>۳</sup> هنر داشتن مجموعه‌ای از گره‌های پردازشی قدرتمند است که به عنوان یک خوشه (یا گروه‌هایی از آنها) به هم متصل شده‌اند تا مجموعه داده‌های عظیم یا مسائل ریاضی پیچیده چند بعدی را به صورت موازی و در سریع‌ترین زمان ممکن حل کنند. بنابراین HPC در مورد محاسبات است. همچنین ممکن است اصطلاح ابررایانه را بشنوید که مترادف خوشه HPC (یا گروه‌هایی از آنها) است. این نام بیشتر به دیدگاه سخت‌افزاری و قدرت آن اشاره دارد که معمولاً به عنوان یک واحد ممیز شناور به نام فلاپ (مانند ترافلاپس، پتافلاپس و ...) محاسبه می‌شود.

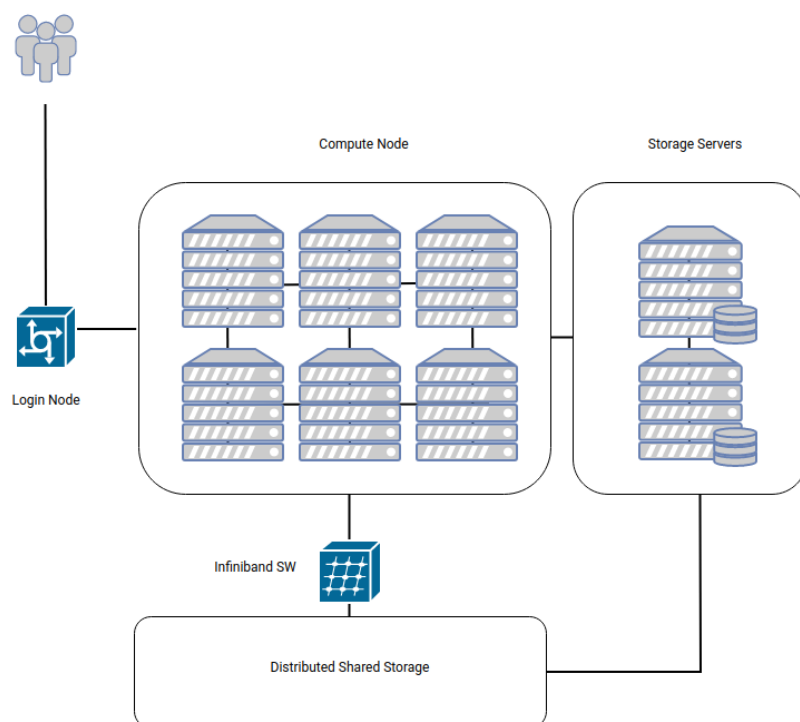
به عنوان مثال، می‌توانید دانش‌آموزی را در نظر بگیرید که یک کد بزرگ (نوشته شده در بیش از هزاران خط) دارد و اگر تمایل داشته باشد کد خود را در لپ‌تاپ شخصی خود اجرا کند، چند سال طول می‌کشد تا پردازش تمام شود و نتیجه نهایی را ببیند؟ بنابراین اگر کد خود را به ابررایانه منتقل کنند، در مدت زمان بسیار کوتاهی به نتایج دست می‌یابد.

<sup>۱</sup>back-end

<sup>۲</sup>National High Performance Computing Center

<sup>۳</sup>High Performance Computing

## ۲.۲.۱ معماری



شکل ۱.۱: اجزای یک خوشه

معمولاً یک خوشه پایه از این اجزای اصلی تشکیل شده است:

- یک گره ورود (گره سر)
- انواع گره‌های محاسباتی
- یک فضای ذخیره‌سازی مشترک توزیع‌شده
- سرورهای GPU (برای بهره‌مندی از محاسبات گرافیکی عظیم)
- یک سوئیچ Infiniband برای اتصال گره‌ها
- یک زمان‌بند کار و یک سیستم صف‌بندی
- یک گره انتقال
- یک گره پس‌پردازش

## ۳.۲.۱ نحوه کار

کاربران به گره ورود (گره سر) وارد می‌شوند و فایل‌های ورودی خود را آپلود می‌کنند. سپس کارهای خود را به زمان‌بند ارسال می‌کنند و در صف انتظار قرار می‌گیرند. سپس زمان‌بند شروع به ارسال کارهای آنها به تمام گره‌های محاسباتی دیگری که کاربر مجاز به دسترسی به آنها است، می‌کند.

## ۳.۱ پروژه

در برخی از موارد مرکز نیاز دارد تا سرورهای مجازی ویندوزی را در اختیار مشتریان خود بگذارد. دلیل استفاده از ویندوز راحتی و سادگی بیشتر برای دانشجویان و مشتریانی است که رشته و تخصص آنها در حوزه کامپیوتر نیست و نیاز دارند تا از قدرت محاسباتی مرکز استفاده کنند. برای این افراد کار با سیستم عامل Linux که به طور معمول در خوشه ها استفاده میشود بسیار دشوار است.

نیاز دیگر این بود که اختصاص سرور ها به مشتریان به صورت خودکار و با کمترین مداخله ممکن از طرف یک نیروی انسانی صورت بگیرد. به همین دلیل استفاده از ابزار هایی مانند Proxmox که در این مرکز استفاده می شود ممکن نبود. از طرفی در دسترس بودن و سلامتی این سرور ها مهم ترین رکن آنها است. در واقع این پردازش ها به دلیل زمان بر بودن آنها از اهمیت خاصی برخوردار هستند. به عنوان مثال در صورتی که سرور میزبان دچار خطا شود، سرور مجازی ویندوز نیز از کار می افتد و تمام منابع و زمانی که برای پردازش استفاده شده بود به هدر می رود. از دید یک مصرف کننده این اتفاق کاملاً غیر قابل تحمل است زیرا که ممکن است مصرف کننده زمان لازم برای این اجرای دوباره این پردازش را نداشته باشد. هدف این پروژه برطرف سازی این مشکلات و نیاز ها بود. در کنار این ها قابلیت هایی مانند تخصیص منابع به صورت زنده، اضافه کردن و حذف دیسک ها به صورت زنده و برقرار سازی سطح مناسبی از امنیت از اهداف مهم دیگر این پروژه بودند.

## فصل ۲

# آماده سازی

### ۱.۲ آماده سازی کامپیوترها

### ۲.۲ آماده سازی سرورها

در این بخش، به بررسی روش‌های آماده‌سازی سرورها و کامپیوترها می‌پردازیم. این فرآیند شامل نصب سیستم‌عامل، ابزارهای مدیریت و پیکربندی شبکه است. در ادامه، به بررسی روش‌های آماده‌سازی سرورها و کامپیوترها می‌پردازیم. این فرآیند شامل نصب سیستم‌عامل، ابزارهای مدیریت و پیکربندی شبکه است. در ادامه، به بررسی روش‌های آماده‌سازی سرورها و کامپیوترها می‌پردازیم. این فرآیند شامل نصب سیستم‌عامل، ابزارهای مدیریت و پیکربندی شبکه است.

## فصل ۳

# ارکستراسیون

### Kubernetes ۱.۳

در این فصل، ما به بررسی معماری و اجزای اصلی Kubernetes می‌پردازیم. ابتدا با مفهوم Pod شروع می‌کنیم، که کوچکترین واحد در Kubernetes است. سپس به Service و Ingress می‌پردازیم که برای مدیریت ترافیک و دسترسی به سرویس‌ها استفاده می‌شوند. در ادامه، به Storage و ConfigMap می‌پردازیم که برای مدیریت داده‌ها و پیکربندی استفاده می‌شوند. در نهایت، به Helm می‌پردازیم که یک ابزار برای مدیریت بسته‌های Kubernetes است.

در این فصل، ما به بررسی معماری و اجزای اصلی Kubernetes می‌پردازیم. ابتدا با مفهوم Pod شروع می‌کنیم، که کوچکترین واحد در Kubernetes است. سپس به Service و Ingress می‌پردازیم که برای مدیریت ترافیک و دسترسی به سرویس‌ها استفاده می‌شوند. در ادامه، به Storage و ConfigMap می‌پردازیم که برای مدیریت داده‌ها و پیکربندی استفاده می‌شوند. در نهایت، به Helm می‌پردازیم که یک ابزار برای مدیریت بسته‌های Kubernetes است.

در این فصل، ما به بررسی معماری و اجزای اصلی Kubernetes می‌پردازیم. ابتدا با مفهوم Pod شروع می‌کنیم، که کوچکترین واحد در Kubernetes است. سپس به Service و Ingress می‌پردازیم که برای مدیریت ترافیک و دسترسی به سرویس‌ها استفاده می‌شوند. در ادامه، به Storage و ConfigMap می‌پردازیم که برای مدیریت داده‌ها و پیکربندی استفاده می‌شوند. در نهایت، به Helm می‌پردازیم که یک ابزار برای مدیریت بسته‌های Kubernetes است.

در این فصل، ما به بررسی معماری و اجزای اصلی Kubernetes می‌پردازیم. ابتدا با مفهوم Pod شروع می‌کنیم، که کوچکترین واحد در Kubernetes است. سپس به Service و Ingress می‌پردازیم که برای مدیریت ترافیک و دسترسی به سرویس‌ها استفاده می‌شوند. در ادامه، به Storage و ConfigMap می‌پردازیم که برای مدیریت داده‌ها و پیکربندی استفاده می‌شوند. در نهایت، به Helm می‌پردازیم که یک ابزار برای مدیریت بسته‌های Kubernetes است.



000000.000 000000 000000.000 000000 000000 0000000000

0000. 000 000 0000 000000000000 000 000000, 000000 000

000000.000 000000 00000000000 0000 0000000. 000 0000000 00000000. 00000000

00000000, 0000000000 0000 00000 0000 0000 00, 00, 0000000000 0000, 00000000 00,

## فصل ۴

# پیاده‌سازی شبکه

**calico** ۱.۴

**multus** ۲.۴

**macvtap ۳.۴**

**bridge** ۴.۴

## kubemacpool ৫.২

0000000000 0000 00000. 00 0000 000000. 00000 00 0000. 00 00000000 0000. 000000000 00000000 0000  
 0000 00 00000. 00000 000000 0000000000 0000 0000. 00 000000 00000. 0000 00000 0000. 00000000000  
 0000000 00000 00 000000 0000 0000000 00000. 0000 0000000000000000 0000000000. 00000000 0000000000  
 0000000 00000. 00. 00000000 0. 0000000000 0000. 0000000000000000 00000. 0000000 00000000 00000000000.  
 0000000. 0000000000 0000 00000 00000000. 00000000000 00 00000 000000000000000. 0000000 0000000. 0000  
 00000000000 00000000 00000000 0000000 00000. 00 000000 00000000. 000000000000 000000 0000000 0000000  
 00000000 0000000000 00000000000. 00000000000 0000 0000000. 000000000 0000 0000000000 0000000 00  
 000000. 00. 0000000000 0000000000. 0000000000 0000. 000000000000 0000000.

## فصل ۵

# بستر مجازی سازی در k8s

### ۱.۵ Kubevirt

### ۲.۵ منابع ماشین ها

### ۳.۵ انتقال زنده

### ۴.۵ تخصیص منابع به صورت زنده

در این فصل، ما به بررسی نحوه اجرای ماشین های مجازی در کلاستر k8s می پردازیم. برای این منظور، ما از Kubevirt استفاده می کنیم. Kubevirt یک پکیج است که به شما امکان می دهد ماشین های مجازی را در کلاستر k8s اجرا کنید. در این فصل، ما به بررسی نحوه نصب Kubevirt، نحوه ایجاد ماشین های مجازی، نحوه تخصیص منابع به ماشین های مجازی و نحوه انتقال زنده ماشین های مجازی می پردازیم.

## فصل ۶

## Windows image

## ۱.۶ گرفتن اطلاعات

## ۲.۶ تنظیمات اضافی

00000000 000 00000. 000000000 000 00000 00000000. 000000000 00000000 0000 00000000. 000000  
 00000000000 000000. 00000000 0. 000000000000000 000000. 0000000 00000000 0000. 0000 0000 0000  
 0000000000 00000000000. 000000000000000 0000 000000 0000000 000 0000000000000 000000. 00.  
 0000000 000 0000 00 000 000000000 000000. 0000000000 00 0000. 0000 000 0000 00000000000  
 000000000 00 0000000 000000 0000 000000000000 0000. 000000000 00000000 00000000000 00000000.  
 000 00 0000 0000 000 0000000000000 000000. 00000000 00000000 000000000 00 000000 0000  
 000000 0000 00000000. 00000000000 000 0000 0000000000. 00000000 000000000000000. 0000000000  
 000000. 00 0000 00000000000 00000000000 000000. 0 00000 0000 00000000 00 0000000. 0000000000  
 000000000 0000000000. 00000000000 00000 00 0000 00 000000 0000000. 000000 0000000. 000000000000  
 000000000000000 0000. 00000000000000 0000000 0000 0000000. 000000000 0000000000 0000 0000000.  
 000000. 000 000000 000000. 000 000000 000000 0000000000

## فصل ٧

# kubevirt-manager

## فصل ۸

### نظارت

#### Prometheus ۱.۸

#### Graphana ۲.۸

در این فصل، شما با ابزارهای نظارتی آشنا می‌شوید. Prometheus یک سیستم نظارتی متن‌باز است که برای جمع‌آوری و ذخیره‌سازی داده‌های متریک طراحی شده است. Graphana یک ابزار برای بصری‌سازی داده‌های Prometheus است. شما یاد خواهید گرفت که چگونه Prometheus را نصب و پیکربندی کنید و چگونه Graphana را برای نمایش داده‌های Prometheus استفاده کنید.

پیوست آ

## اطلاعات اضافی

0000000000 00000000, 000000000000 0000 00000000 00 000000000000 000000000 00000000 0000 0000000000. 0000  
 00, 000000000000 00000, 00000 00000000000000 00000. 0000 0000000 000000000000000000 00 0000. 000000000000  
 00000000. 00000000000 00000000000 00 00000000. 00 00000000 00000000. 00, 000000000 00, 000000000000000000  
 00000000000. 00000 000000000 000000000. 0000000 0000000. 00000 000 00000 0000000000000 00000000000  
 000000. 0000 0000000 000000000000 00000 00000000. 0000 00000000 0000000000. 000000000