



دانشگاه شهید مدنی آذربایجان

دانشکده کامپیوتر

عنوان پروژه

مقایسه عملکرد کرنل لینوکس در دو کانتینر ساز Docker و Podman

مشخصات دانشجو

- نام نام خانوادگی : محمد مهدی رسول امینی
- شماره دانشجویی : 4001830235
- مقطع تحصیلی : کارشناسی
- رشته تحصیلی : مهندسی کامپیوتر-هوش مصنوعی

مشخصات استاد راهنما

- نام استاد : دکتر سلطانی

عنوان درس : پروژه عملی کارشناسی

نیمسال تحصیلی : نیمسال اول 1404

مقایسه عملکرد کرنل لینوکس در کانتینرهای Docker و Podman

فهرست مطالب

1. مقدمه
2. اهداف پروژه
3. ابزارهای مورد استفاده
4. مراحل اجرایی
5. معیارهای ارزیابی
6. نتایج اولیه
7. تحلیل نتایج
8. نتیجه گیری
9. منابع مورد نیاز
10. منابع و مراجع

مقدمه

در دنیای فناوری اطلاعات و مجازی سازی، کانتینرها نقش حیاتی در توسعه، استقرار و مدیریت نرم افزارها ایفا می کنند Docker به عنوان پیشگام این فناوری شناخته می شود، اما Podman به عنوان جایگزینی سبک تر و امن تر معرفی شده است. این پروژه به مقایسه عملکرد کرنل لینوکس در این دو پلتفرم می پردازد تا تفاوت های آنها در مدیریت منابع سیستم مشخص شود.

اهداف پروژه

- ارزیابی عملکرد CPU و حافظه: مقایسه مصرف منابع در Docker و Podman.
- تعیین کانتینر ساز بهینه: شناسایی پلتفرمی که کارایی بهتری در استفاده از منابع دارد.

- بررسی تأثیرات کرنل لینوکس: تحلیل مدیریت منابع سیستم در هر دو پلتفرم.
- مقایسه فراخوان‌های سیستم: بررسی تفاوت‌ها در فراخوان‌های سیستم بین دو پلتفرم.

ابزارهای مورد استفاده

ابزار	کاربرد
sysbench	تست عملکرد CPU و حافظه
stress-ng	ایجاد فشار روی CPU و حافظه
vmstat	مانیتورینگ CPU ، حافظه و I/O
htop	مشاهده لحظه‌ای مصرف منابع
strace	ردیابی فراخوان‌های سیستم
netstat	بررسی وضعیت شبکه
lsof	مشاهده فایل‌های باز شده توسط فرآیندها
dstat	مانیتورینگ جامع سیستم
iostat	تحلیل عملکرد دیسک

مراحل اجرایی

1. آماده سازی محیط

- نصب Docker و Podman روی سیستم عامل اوبونتو یا کالی لینوکس.
- ایجاد دو کانتینر یکسان با تصویر پایه Alpine.
- نصب ابزارهای تست شامل sysbench، stress-ng، و سایر ابزارها در هر دو کانتینر.

2. اجرای تست ها

• تست: CPU

- اجرای دستور `sysbench cpu --cpu-max-prime=20000 run` در هر دو کانتینر.
- استفاده از `stress-ng --cpu 4 --timeout 60s` برای ایجاد بار روی CPU.
- تست حافظه:

- اجرای دستور `sysbench memory --memory-block-size=1K --memory-total-size=10G run.`
- استفاده از `stress-ng --vm 1 --vm-bytes 1G --timeout 60s`.
- تست شبکه:

- بررسی پهنای باند و تأخیر با ابزار `netstat`.
- تست فراخوان های سیستم:
- استفاده از `strace` برای ردیابی فراخوان های سیستم.

3. جمع آوری و تحلیل داده ها

- ثبت نتایج هر تست در جداول مقایسه ای.
- استفاده از نمودارها برای نمایش تفاوت های عملکردی.

معیارهای ارزیابی

- کارایی CPU: زمان پاسخ دهی و درصد استفاده از CPU.
- کارایی حافظه: میزان مصرف حافظه و زمان دسترسی.

- فراخوان‌های سیستم: تعداد و نوع فراخوان‌های سیستم.
- عملکرد شبکه: پهنای باند و تأخیر.
- مدیریت فایل‌ها: تعداد فایل‌های باز شده و دسترسی به آن‌ها.

نتایج اولیه

مقایسه عملکرد CPU و حافظه

معیار	Docker	Podman
سرعت پردازش (event/sec)	216.65	215.80
سرعت انتقال حافظه (MiB/sec)	2864.69	2840.12
تأخیر متوسط CPU (ms)	4.61	4.63
تأخیر حافظه (ms)	0.00	0.00

مقایسه معماری و امنیت

ویژگی	Docker	Podman
ساختار	Client-Server (Daemon-based)	Daemon-less
نیاز به Daemon	دارد	ندارد
امنیت Rootless	محدود	بله
سازگاری با Dockerfile	بله	بله

تحلیل نتایج

- عملکرد مشابه: تفاوت‌های عملکردی بین Docker و Podman ناچیز است.
- امنیت بالاتر: **Podman** اجرای بدون Daemon و پشتیبانی از Rootless مزیت‌های امنیتی Podman هستند.
- سربار کمتر: **Podman** معماری Daemon-less باعث کاهش سربار سیستم می‌شود.
- اکوسیستم Docker: **Docker** همچنان از نظر ابزارهای جانبی و یکپارچگی با سرویس‌های ابری پیش‌تاز است.

نتیجه‌گیری

- اگر امنیت و معماری سبک‌تر مهم است، Podman گزینه بهتری است.
- اگر یکپارچگی با اکوسیستم ابری و ابزارهای توسعه اهمیت دارد، Docker مناسب‌تر است.
- انتخاب نهایی به نیازهای خاص پروژه و اولویت‌های تیم توسعه بستگی دارد.

منابع مورد نیاز

- سخت‌افزار: سیستم با حداقل 4 گیگابایت RAM و 2 هسته CPU.
- نرم‌افزار: اوبونتو یا کالی لینوکس، Docker، Podman، و ابزارهای تست.
- دسترسی به اینترنت: برای نصب ابزارها و کانتینرها.

منابع و مراجع

1. مستندات رسمی Docker: <https://docs.docker.com>

2. مستندات رسمی Podman: <https://podman.io/docs>
3. راهنمای استفاده از sysbench: <https://github.com/akopytov/sysbench>
4. مقالات مقایسه‌ای Docker و Podman در مجلات معتبر فناوری اطلاعات.

محمد مهدی رسول امینی