

# پروپوزال: تست و مقایسه عملکرد کرنل لینوکس در دو کانتینر ساز های Docker و Podman

## 1. عنوان پروژه

مقایسه عملکرد کرنل لینوکس در دو کانتینر ساز Docker و Podman با استفاده از ابزارهای lsof, netstat, strace, htop, vmstat, stress-ng, sysbench

## 2. هدف پروژه

- ارزیابی و مقایسه عملکرد CPU و حافظه کرنل لینوکس در دو محیط کانتینری Docker و Podman.
- تعیین کانتینر سازی که عملکرد بهتری در استفاده از منابع سیستمی (CPU) و حافظه (ارائه می دهد).
- بررسی تأثیرات کرنل لینوکس در مدیریت منابع در هر دو پلتفرم.

## 3. ابزارهای مورد استفاده

- **sysbench**: برای تست عملکرد CPU و حافظه.
- **stress-ng**: برای ایجاد فشار روی CPU و حافظه و بررسی رفتار کرنل.
- **vmstat**: برای مانیتورینگ استفاده از CPU ، حافظه و I/O.
- **htop**: برای مشاهده زنده مصرف منابع سیستم.
- **strace**: برای ردیابی فراخوان های سیستمی و بررسی عملکرد کرنل.
- **netstat**: برای بررسی وضعیت شبکه و اتصالات.
- **lsof**: برای بررسی فایل های باز شده توسط فرآیندها.

## 4. مراحل اجرایی

### 4.1. آماده سازی محیط

- نصب Docker و Podman روی یک سیستم با کرنل لینوکس (ترجیحاً اوبونتو یا کالی لینوکس).
- ایجاد دو کانتینر یکسان (با سیستم عامل پایه یکسان) در Docker و Podman.
- نصب ابزارهای تست (sysbench, stress-ng, vmstat, htop, strace, netstat, lsof) در هر دو کانتینر.

### 4.2. اجرای تست ها

## 1. تست CPU:

- اجرای `sysbench cpu test` در هر دو کانتینر.
- اجرای `stress-ng --cpu` برای ایجاد فشار روی CPU و بررسی رفتار کرنل.
- ثبت نتایج با استفاده از `htop` و `vmstat`.

## 2. تست حافظه:

- اجرای `sysbench memory test` در هر دو کانتینر.
- اجرای `stress-ng --vm` برای ایجاد فشار روی حافظه.
- ثبت نتایج با استفاده از `htop` و `vmstat`.

## 3. بررسی فراخوان‌های سیستمی:

- استفاده از `strace` برای ردیابی فراخوان‌های سیستمی در هر دو کانتینر.
- مقایسه تعداد و نوع فراخوان‌های سیستمی در `Docker` و `Podman`.

## 4. بررسی شبکه:

- استفاده از `netstat` برای بررسی وضعیت شبکه در هر دو کانتینر.
- مقایسه عملکرد شبکه در `Docker` و `Podman`.

## 5. بررسی فایل‌های باز شده:

- استفاده از `lssof` برای بررسی فایل‌های باز شده توسط فرآیندها در هر دو کانتینر.

## 4.3. تحلیل داده‌ها

- مقایسه نتایج تست‌های CPU و حافظه در `Docker` و `Podman`.
- بررسی تفاوت‌ها در فراخوان‌های سیستمی و تأثیر آن‌ها بر عملکرد کرنل.
- تحلیل داده‌های شبکه و فایل‌های باز شده.

## 5. معیارهای ارزیابی

- کارایی CPU: زمان پاسخ‌دهی و استفاده از CPU در هر دو پلتفرم.
- کارایی حافظه: میزان استفاده از حافظه و زمان دسترسی به آن.
- فراخوان‌های سیستمی: تعداد و نوع فراخوان‌های سیستمی در هر دو کانتینر.
- عملکرد شبکه: تأخیر و پهنای باند در هر دو پلتفرم.
- مدیریت فایل‌ها: تعداد فایل‌های باز شده و دسترسی به آن‌ها.

## 6. گزارش‌دهی

- ارائه جدول‌ها و نمودارهای مقایسه‌ای برای هر یک از معیارهای ارزیابی.
- تحلیل نتایج و تعیین کانتینر سازی که عملکرد بهتری در استفاده از منابع کرنل لینوکس ارائه می‌دهد.
- ارائه پیشنهادات برای بهبود عملکرد در هر دو پلتفرم.

## 7. منابع مورد نیاز

- یک سیستم با کرنل لینوکس (ترجیحاً اوبونتو یا کالی لینوکس).
- دسترسی به اینترنت برای نصب ابزارها و کانتینرها.
- حداقل 4 گیگابایت RAM و 2 هسته CPU برای اجرای تست‌ها.

محمد مهدی رسول امینی