# **بخش اول**

## الف)

خیلی خلاصه بخوایم بگیم، مجازی‌سازی می‌شه استفاده بهینه از سخت‌افزار به صورتی که به کمک‌ آن می‌توانیم نمونه‌هایی مجازی از زیرساخت یا سخت‌افزار یا حافظه‌ یا واحدهای ذخیره‌سازی و... ایجاد کنیم. مجازی‌سازی توانسته ارتباط میان سخت‌افزار و نرم‌افزار را دگرگون‌کرده و به عنوان یکی از مبانی اصلی رایانش ابری شناخته می‌شود که به استفاده بهینه و تمام و کمال از ظرفیت‌های رایانش ابری کمک می‌کند.

## ب)

معماری چند اجاره‌ای در رایانش ابری به این معناست که در آن مشتریان مختلف از منابع یکسانی استفاده می‌کنند به علاوه اطلاعات آن‌ها از یک‌دیگر ایزوله است و مشتریان به اطلاعات بقیه دسترسی‌ای ندارند.

مشکلاتی که این معماری دارد عبارتند از:

امنیت: درسته که کاربران اطلاعات هم‌دیگر را نمی‌بینند اما دارند از منابع مشترکی استفاده می‌کنند که می‌تواند امنیت را به خطر بیاندازد.

قدرت: هرچقدر هم که این قابلیت به خوبی پیاده شده باشد نمی‌تواند به قدرت حالتی که به صورت اختصاصی شما به منابع دسترسی دارید، باشد.

مدیریت و دسترسی: در این حالت شما دسترسی کامل و تمام را بر منابع ندارید و کمی محدود شده‌اید.

## ج)

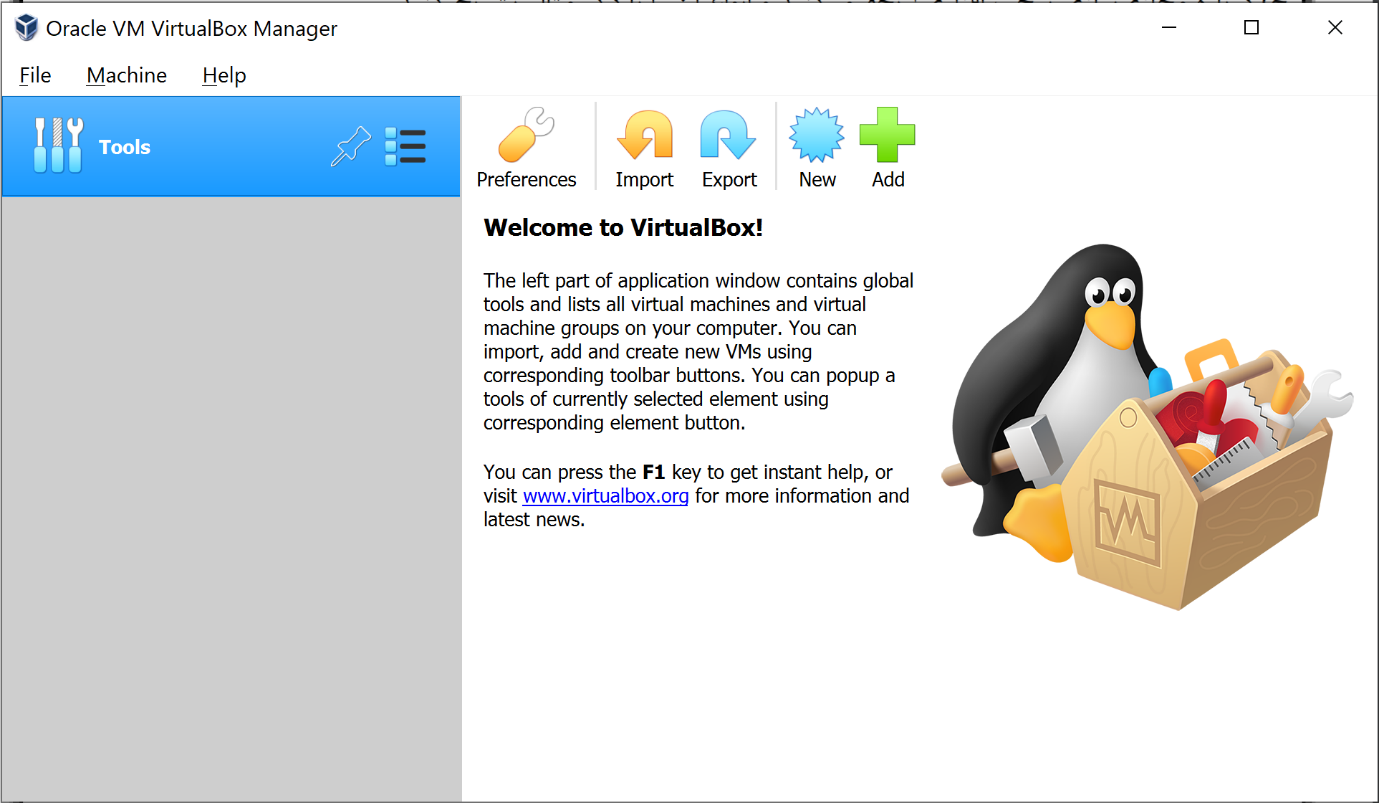
در مجازی‌سازی سخت‌افزاری از hypervisor استفاده می‌شود که یک لایه‌ میان سخت‌افزار و نرم‌افزار است و منابع فیزیکی را مدیریت می‌کند و ساختارهایی مجازی مانند پردازنده‌های مجازی را به وجود می‌آورد که این مجازی‌سازی سخت‌افزاری نیز انواعی دارد که عبارتند از:

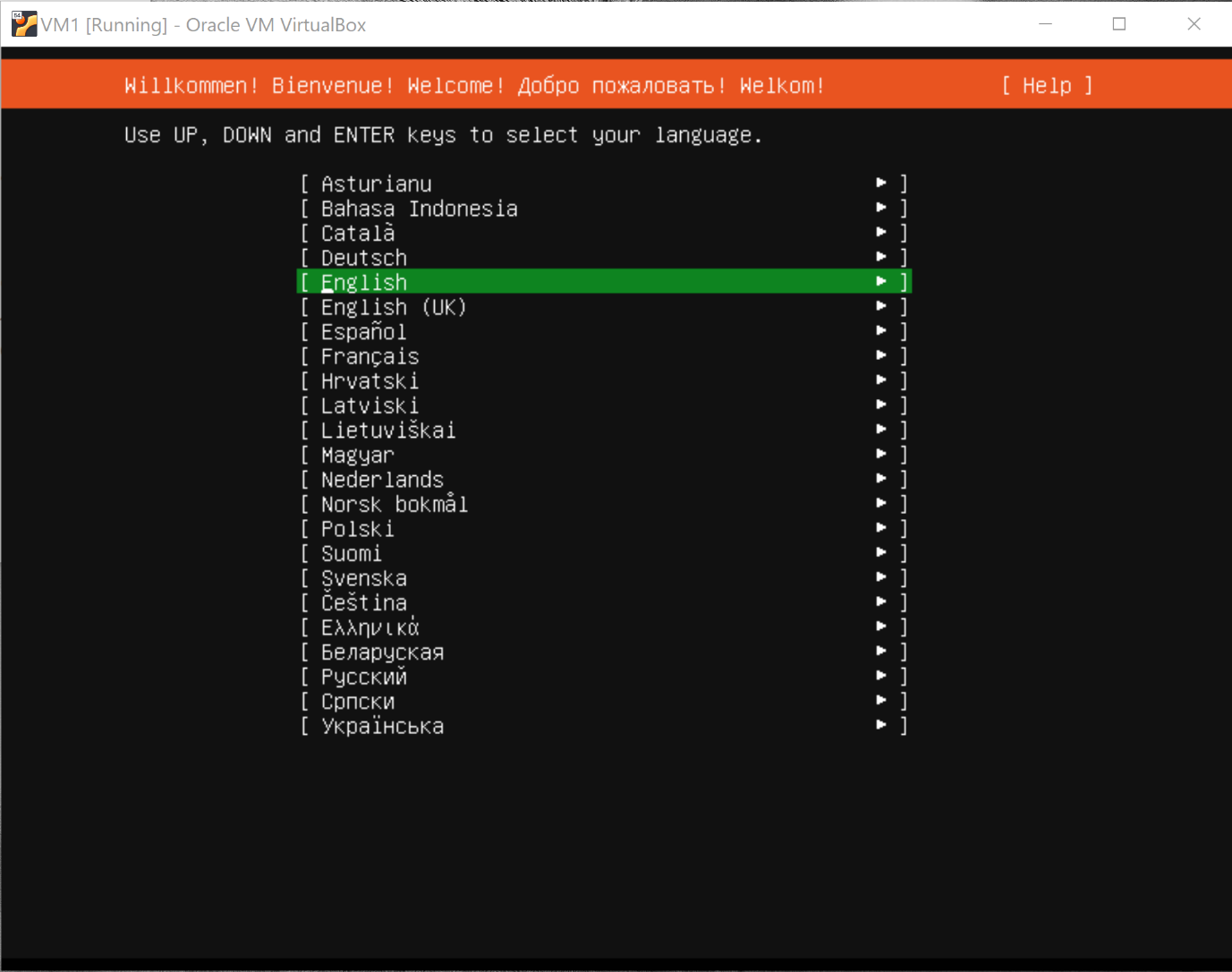
مجازی‌سازی کامل: در این حالت هیچ تغییری برای اجرای برنامه‌ها لازم نیست انجام شود و معماری سخت‌افزاری به طور کامل شبیه‌سازی شده است.

مجازی‌سازی شبیه‌سازی: در این حالت سخت‌افزار توسط ماشین مجازی شبیه‌سازی می‌شود.

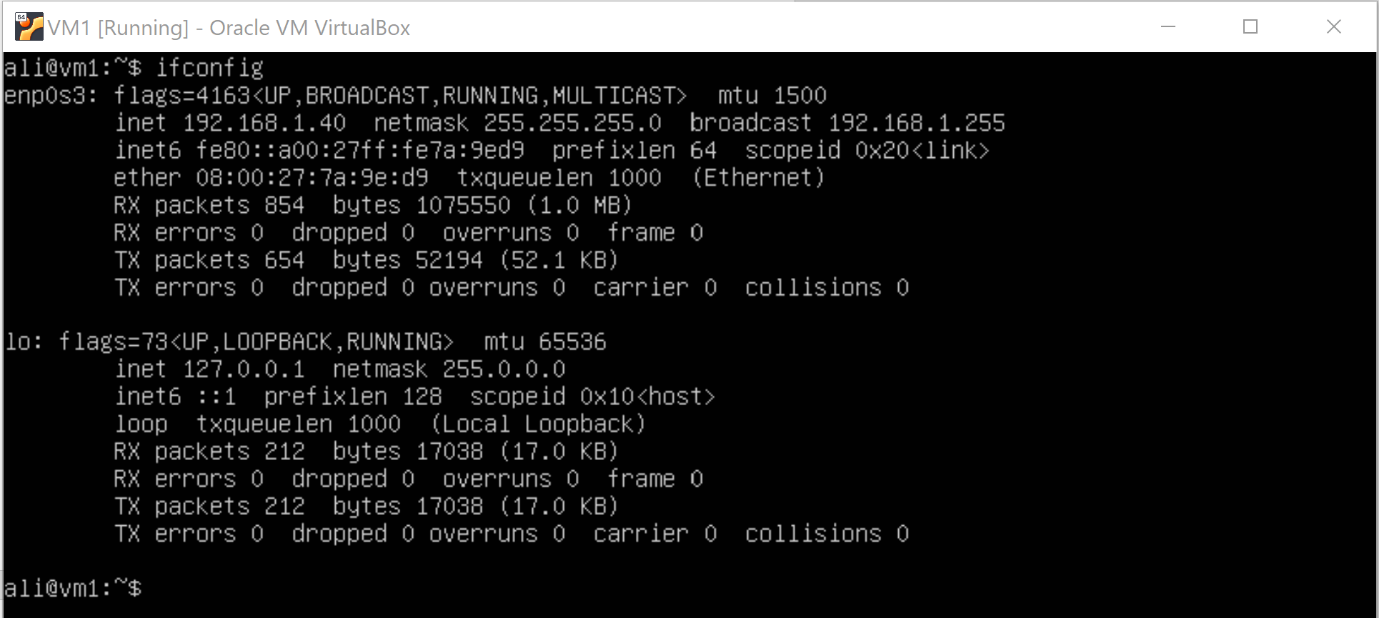
نیمه مجازی‌سازی: در این حالت سخت‌افزار اصلا شبیه‌سازی نمی‌شود.

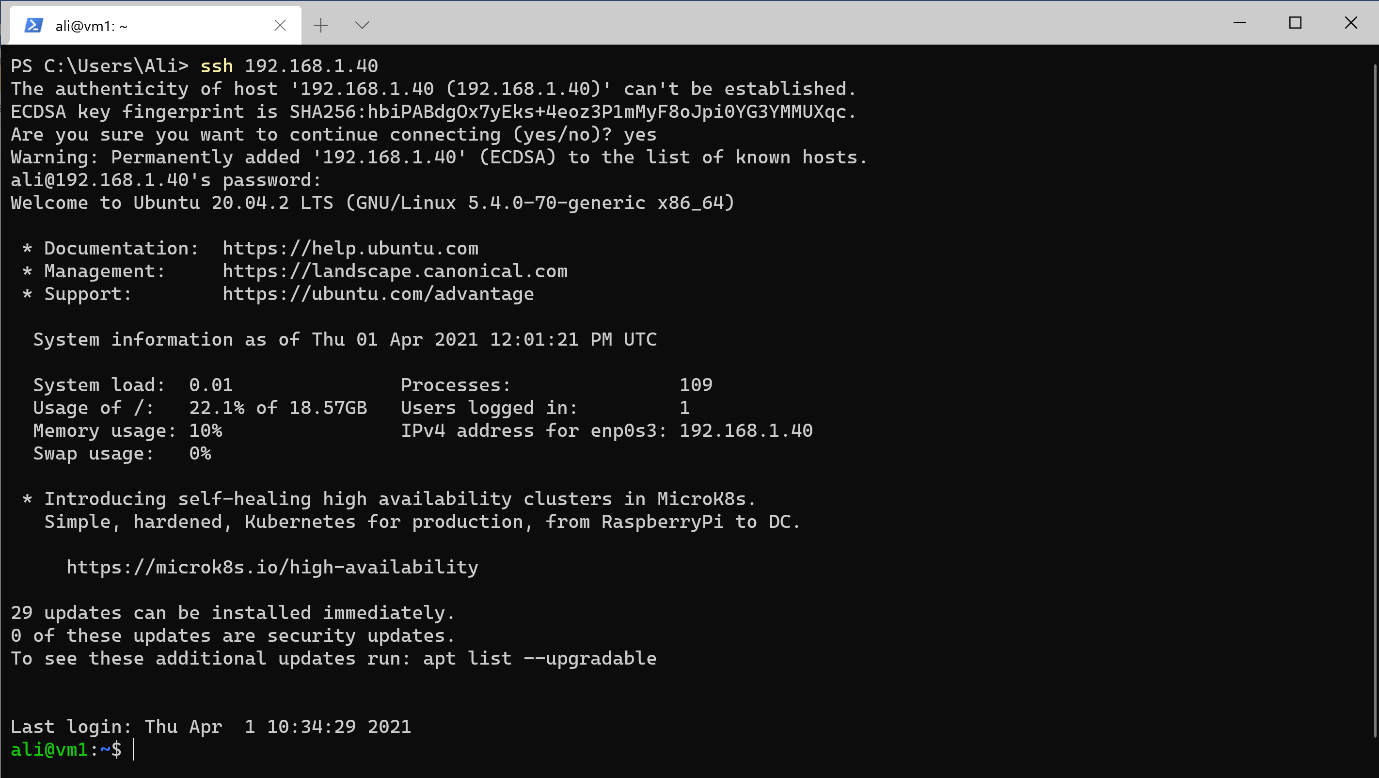
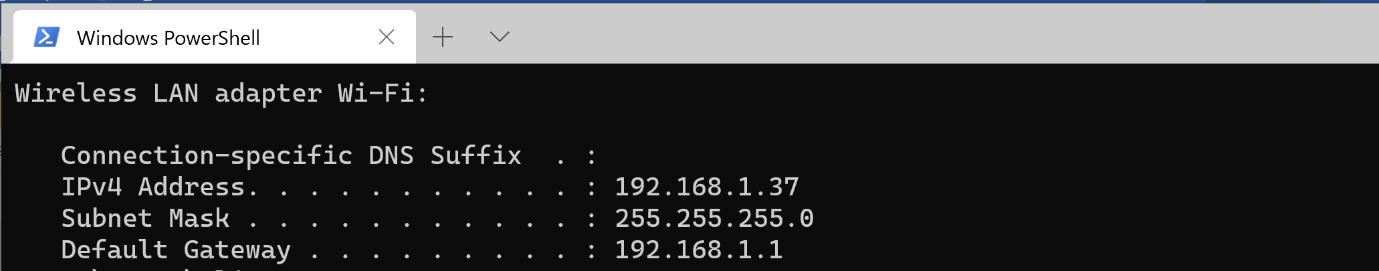
# **بخش دوم**

ابتدا نرم‌افزار virtualbox را نصب می‌کنیم.

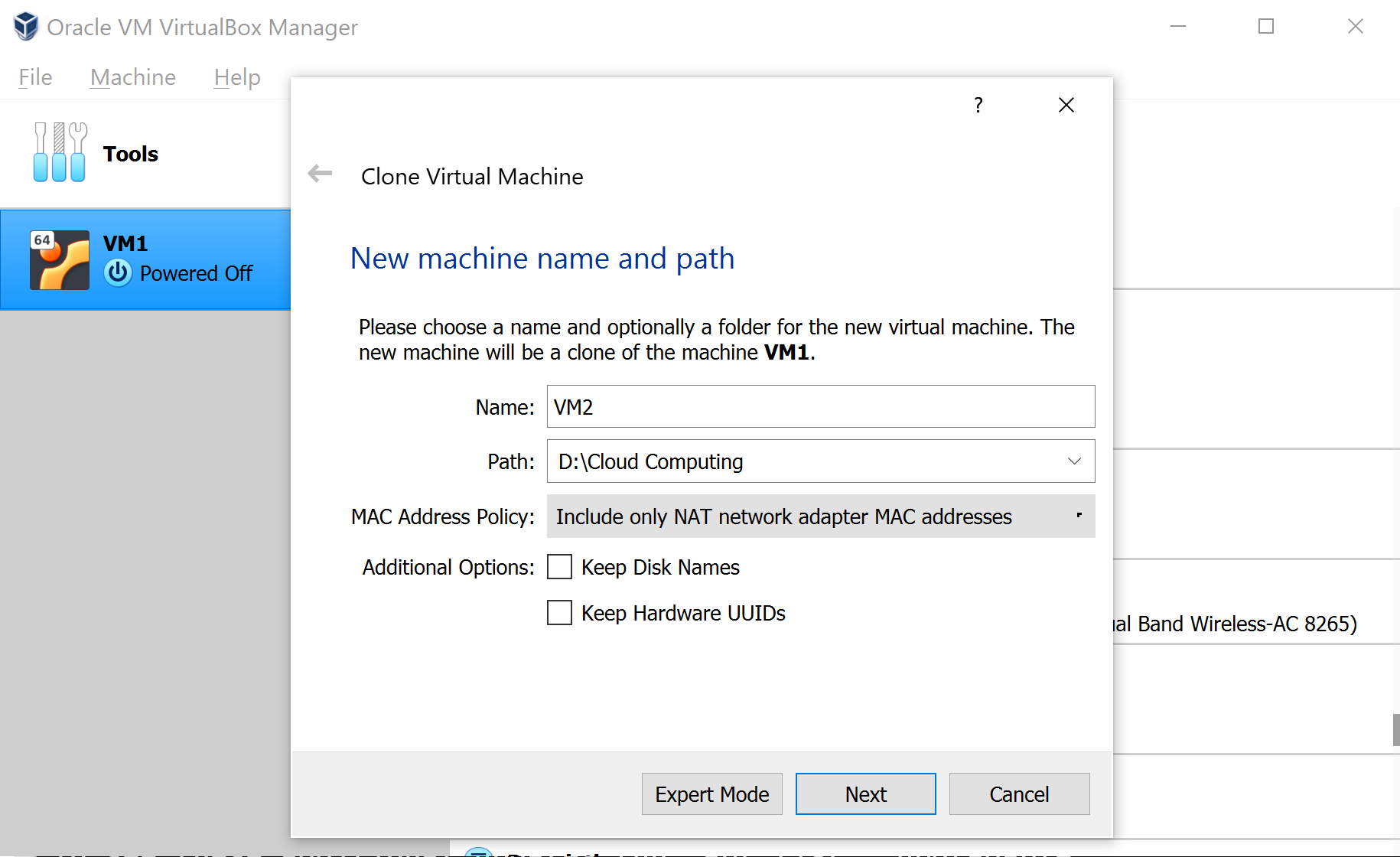
سپس با توجه به اسلاید‌های مربوط به ایجاد ماشین مجازی، شروع به ایجاد ماشین مجازی VM1 می‌کنیم:

در نهایت این ماشین آماده می‌شود:

با قرار دادن شبکه‌ی این ماشین بر روی bridge یک ip به این ماشین اختصاص می‌یابد:

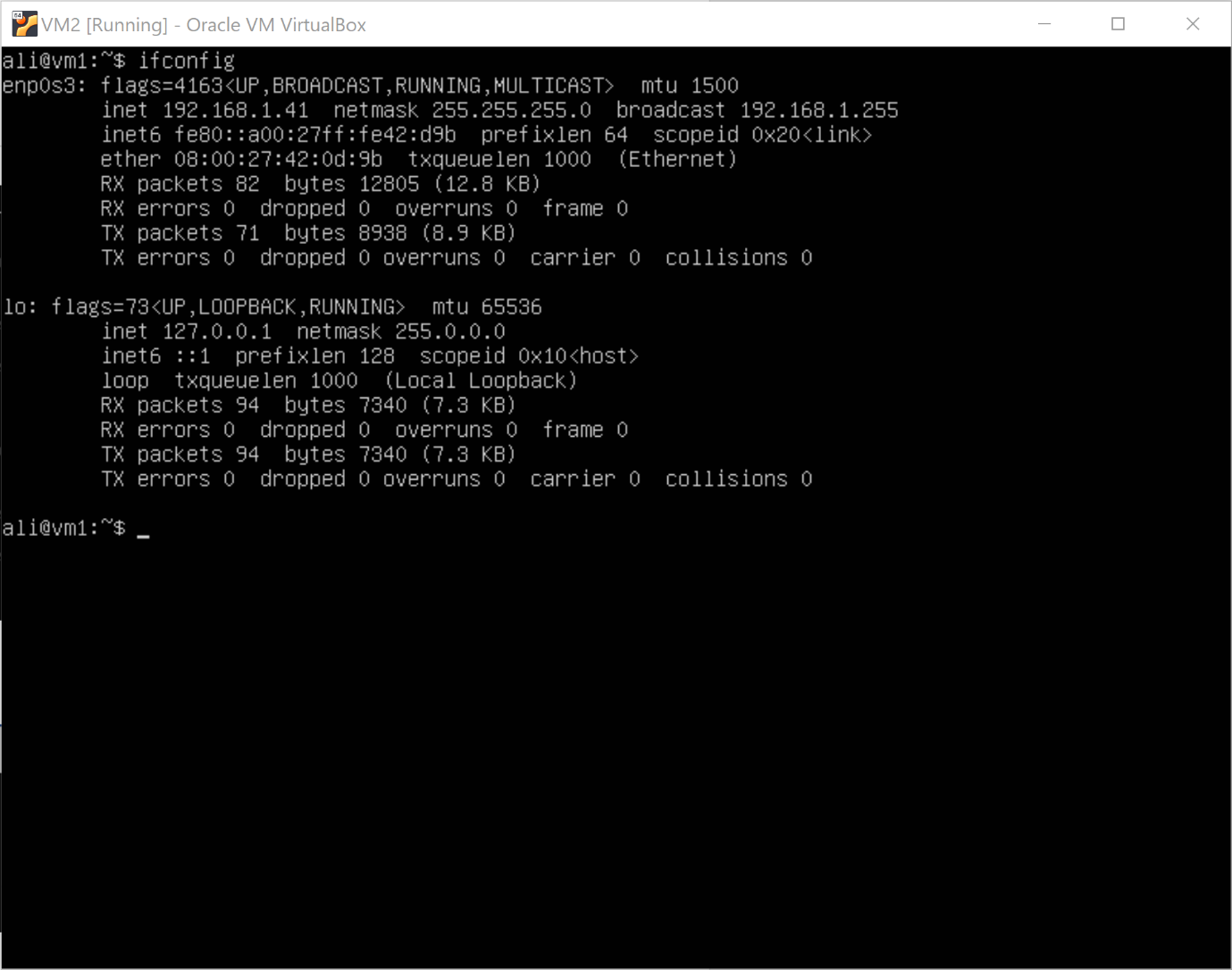
همان‌طور که می‌بینیم ماشین مجازی با کامپیوتر میزبان در یک ساب‌نت هستند:

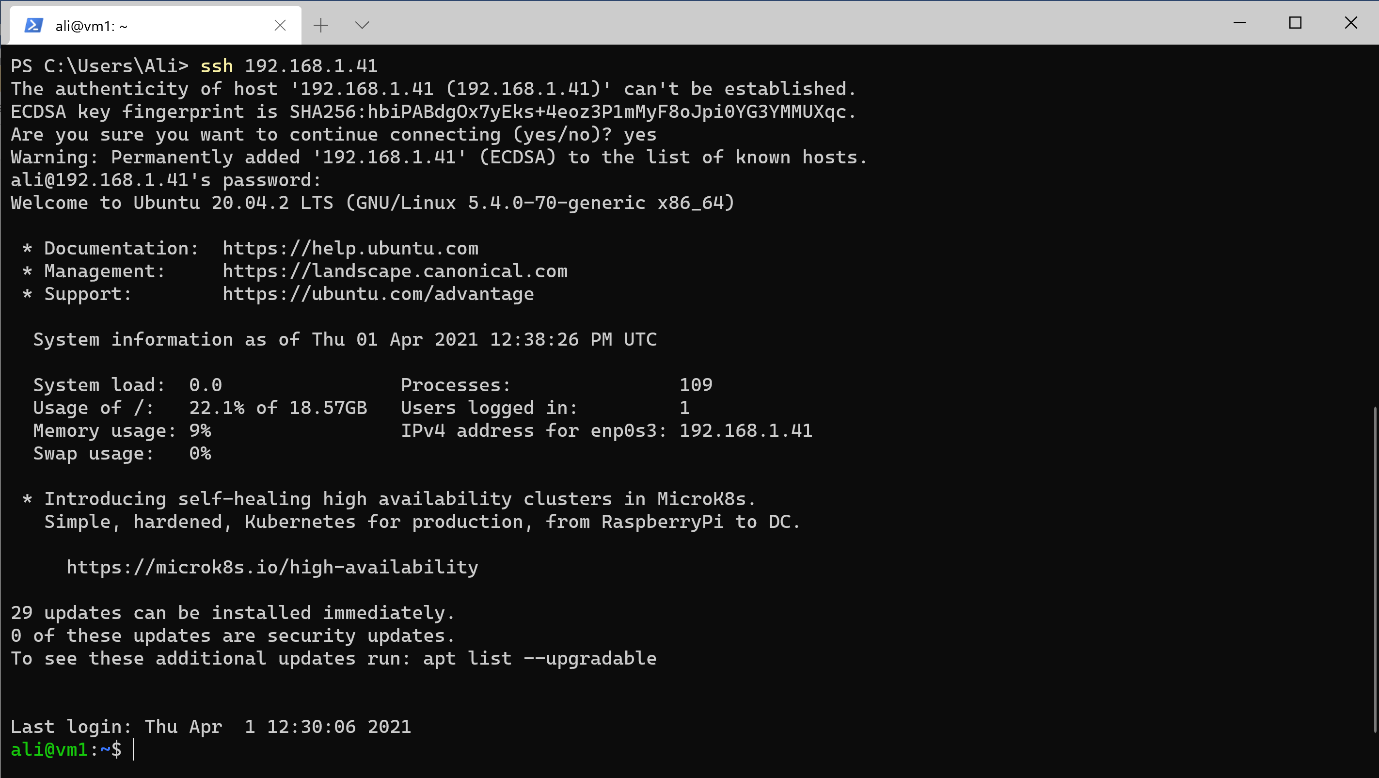
می‌توانیم به این ماشین از طریق ssh دسترسی داشته باشیم:

سپس ماشین VM2 را از طریق clone کردن ماشین اولی ایجاد می‌کنیم:

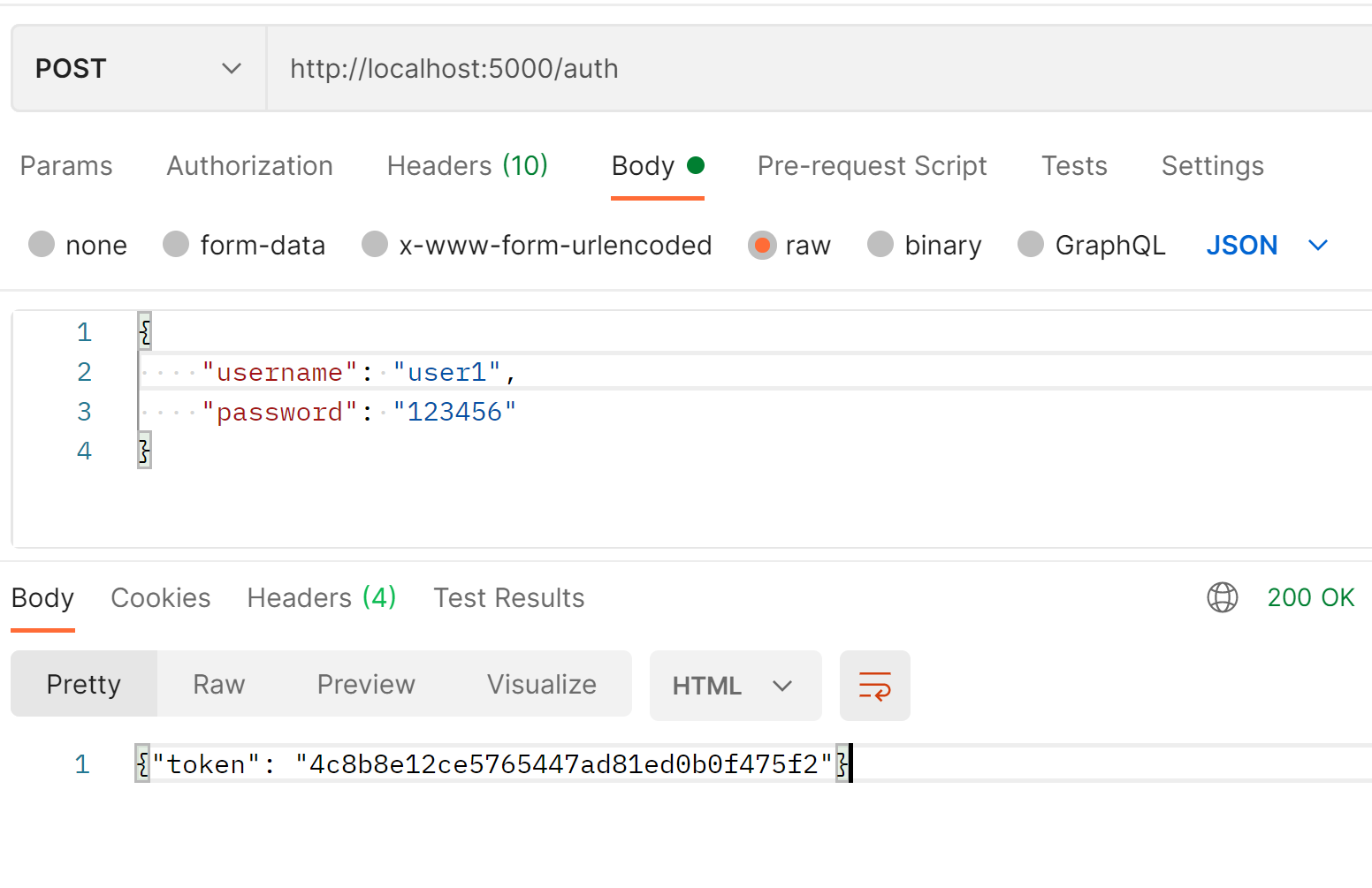
با قرار دادن شبکه‌ی این ماشین نیز بر روی bridge برای آن یک ip می‌گیریم:

sudo ip address flush scope global dynamic

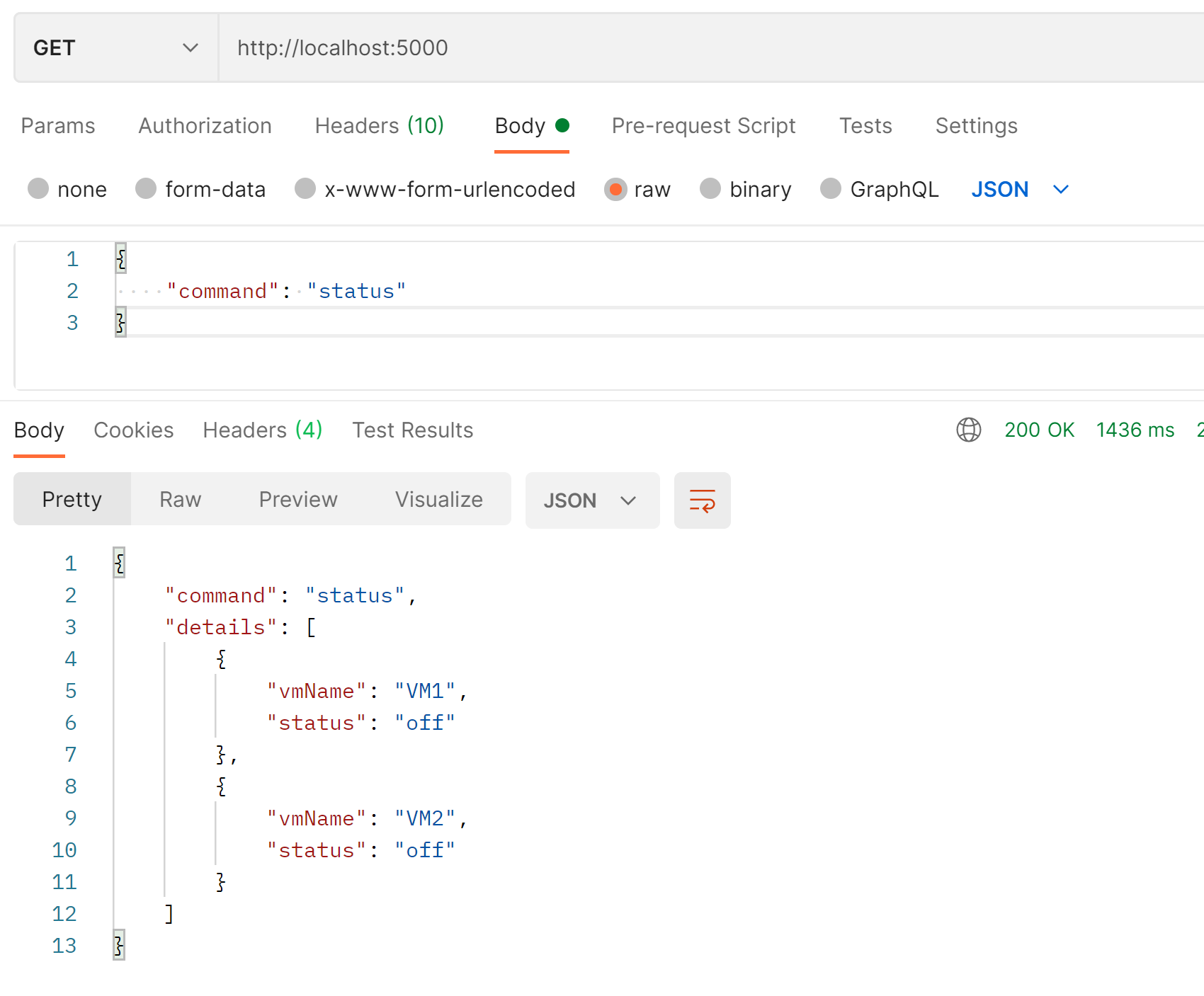
sudo dhclient -v

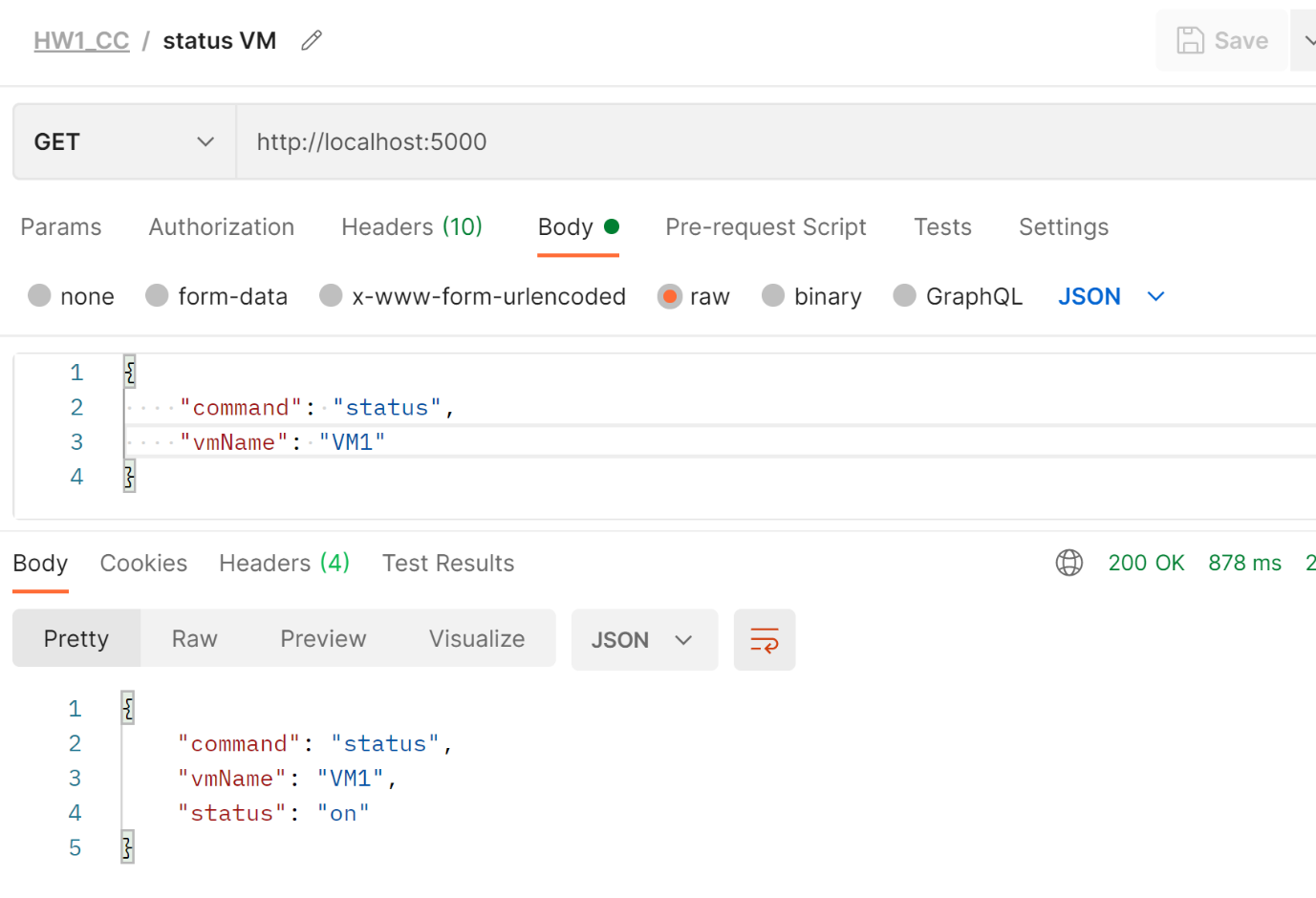
و از طریق ssh می‌توانیم به این ماشین متصل شویم:

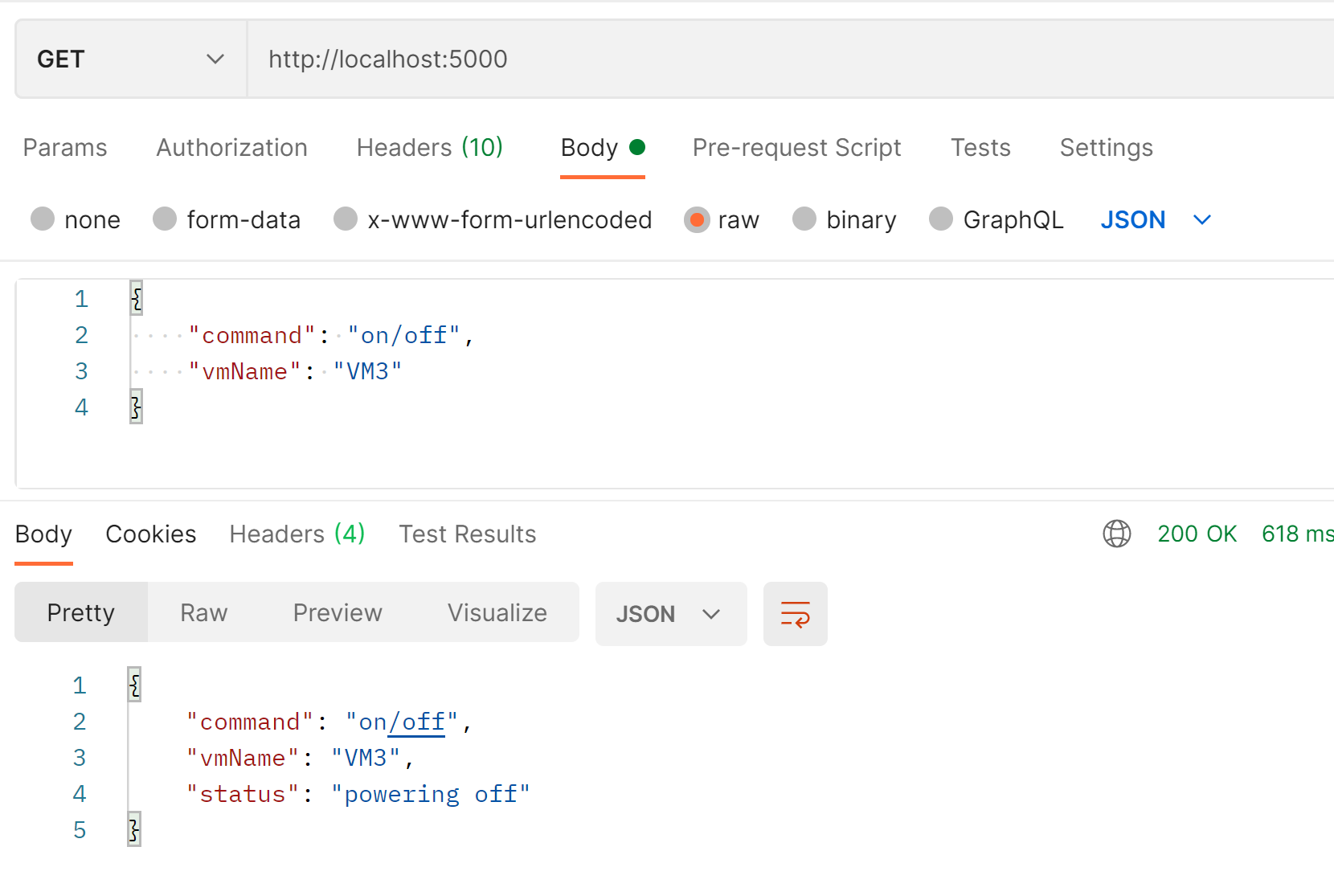
حال از طریق سرور پایتونی‌ای که فراهم کردیم به سراغ مدیریت سرورها می‌رویم. این سرور به کمک زبان پایتون و کتاب‌خانه‌ی flask فراهم شده است و برای پیاده‌سازی ارتباطات لازم با ماشین‌ها هم در تعدادی از نیازمندی‌ها از یکی از کتاب‌خانه‌های ارتباطی پایتون و virtualbox استفاده شده و مواردی که در این کتاب‌خانه موجود نبودند نیز با کمک دستورات خود virtualbox یعنی vboxmanage انجام شده‌ است.

در ابتدا به صورت زیر باید در سیستم به عنوان یوزر admin یا user1 لاگین کنیم:

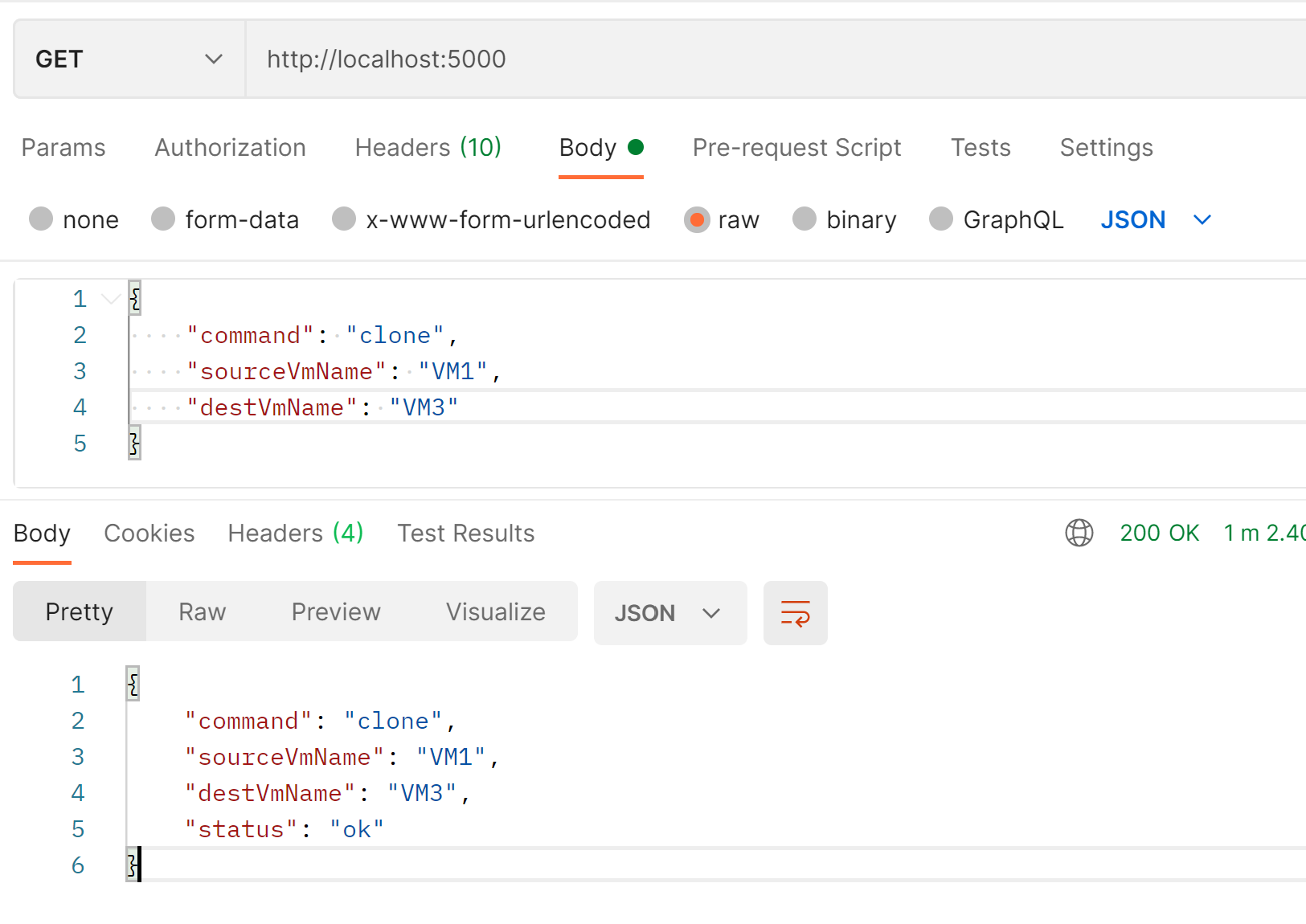
در قسمت اول می‌توانیم وضعیت تمامی ماشین‌ها را ببینیم:



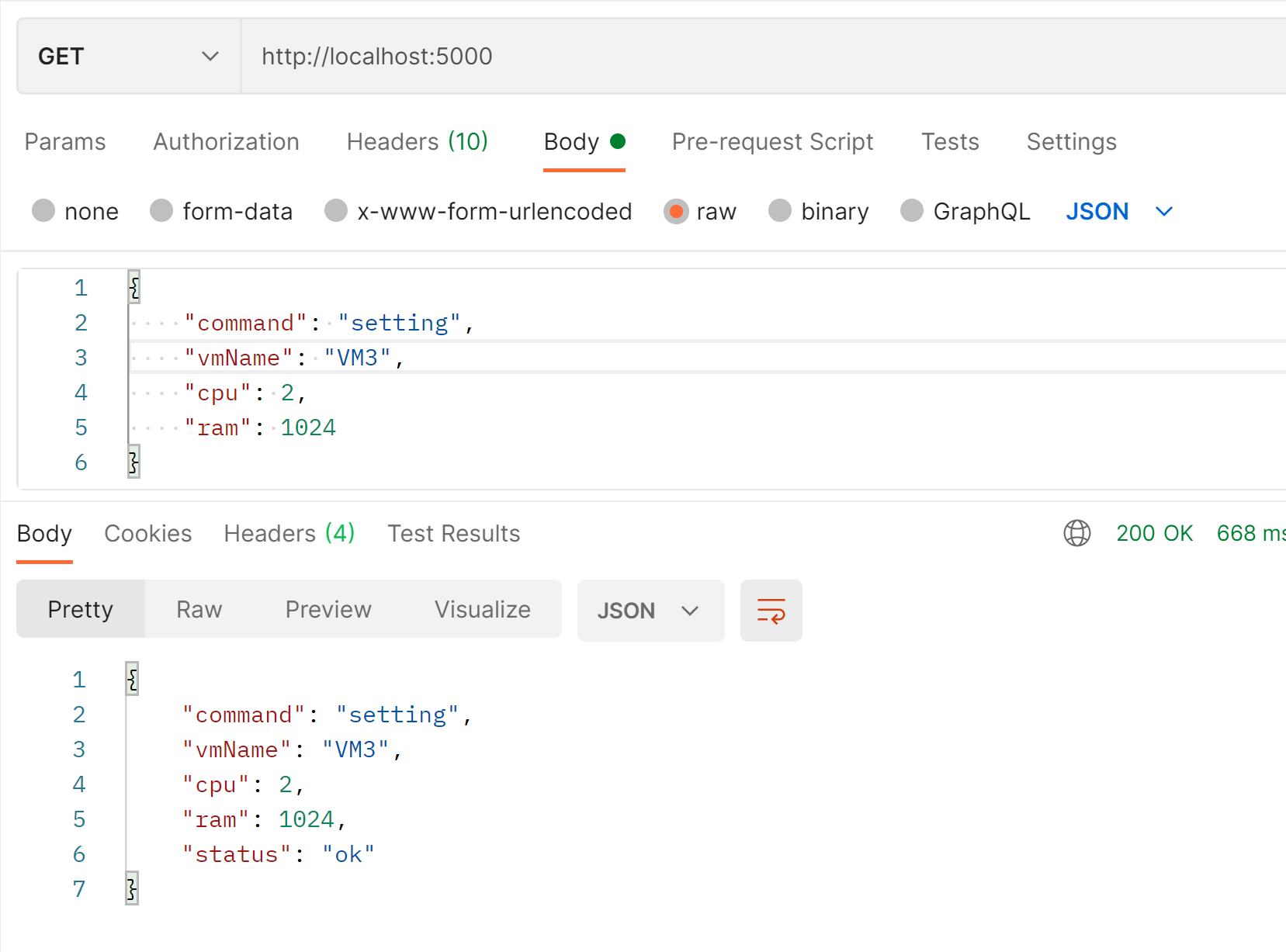
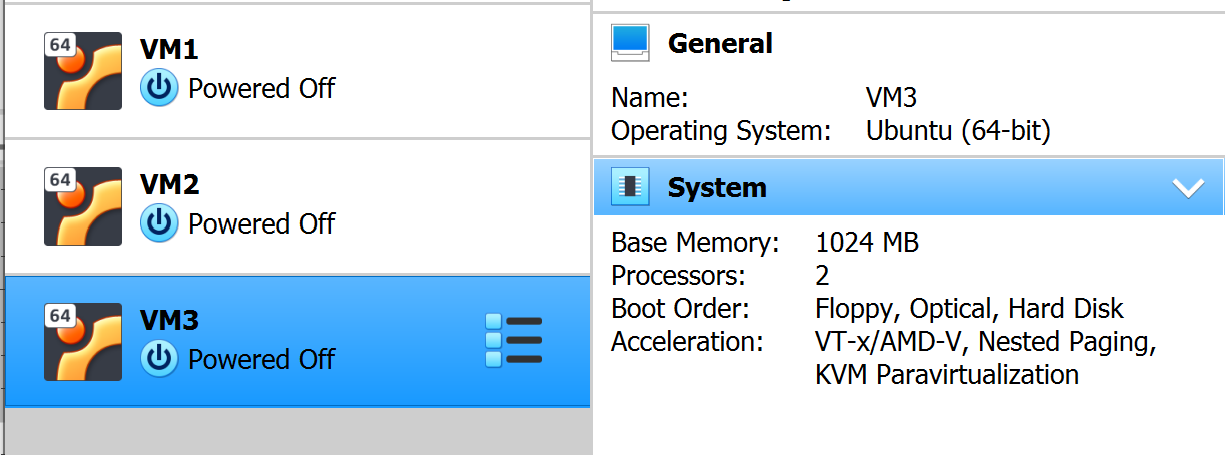
در این قسمت می‌توانیم وضعیت یک ماشین خاص را ببینیم:

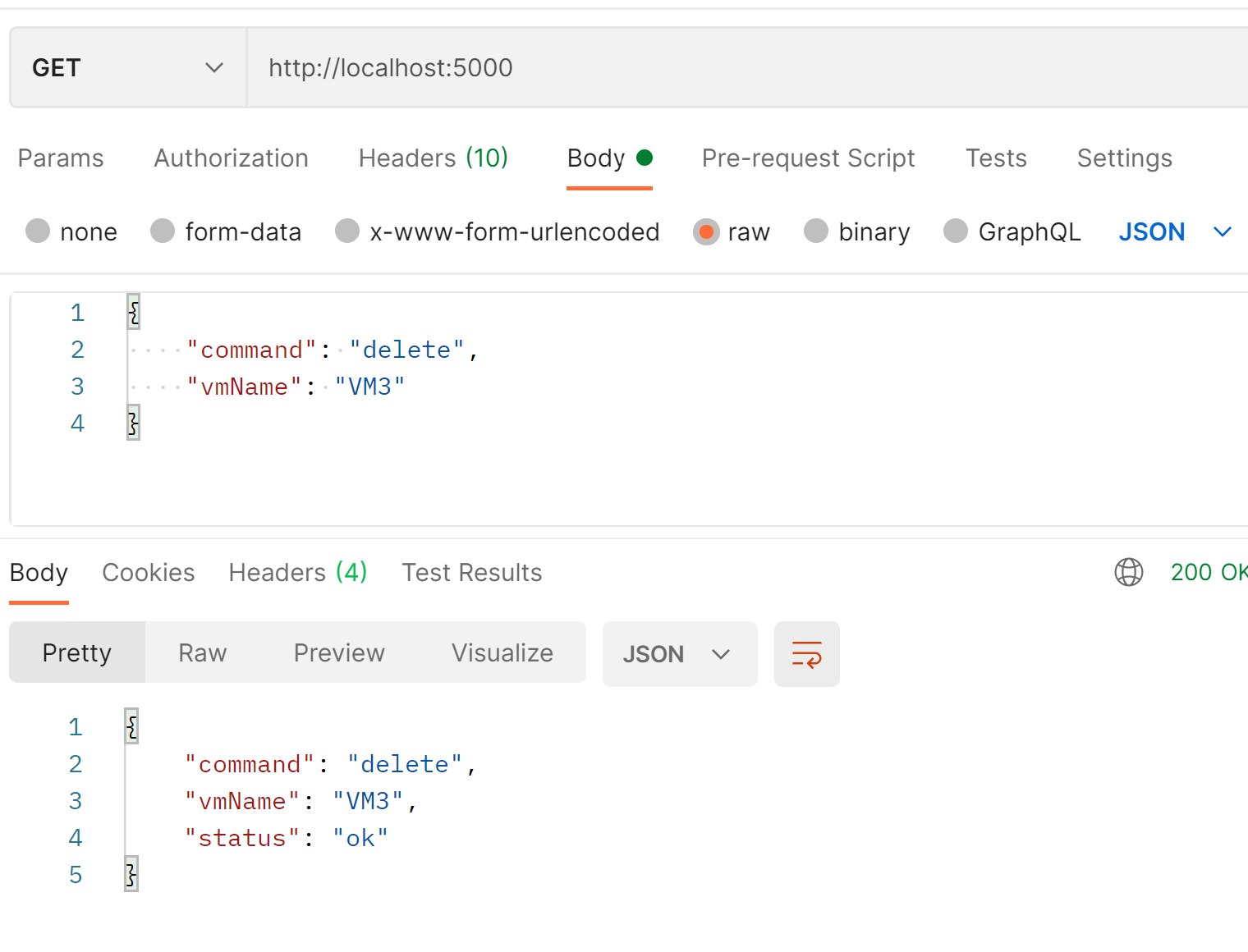
 در این قسمت می‌توانیم یک ماشین را خاموش یا روشن کنیم:

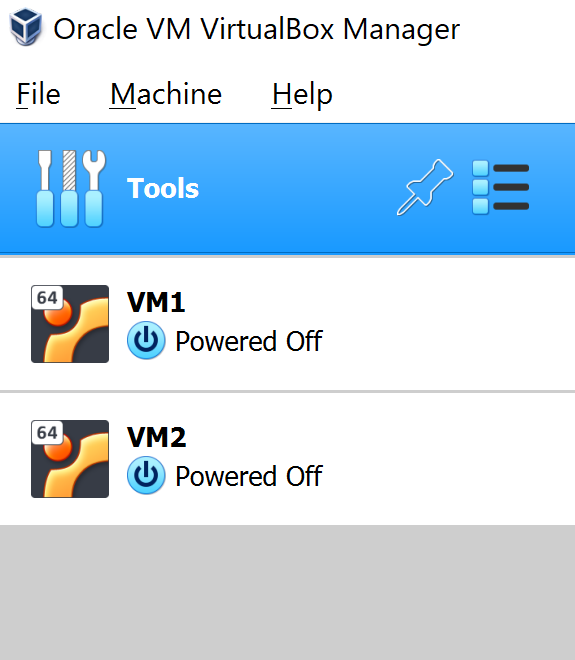
در این قسمت می‌توانیم یک ماشین جدید را از روی یک ماشین قدیمی ایجاد کنیم:



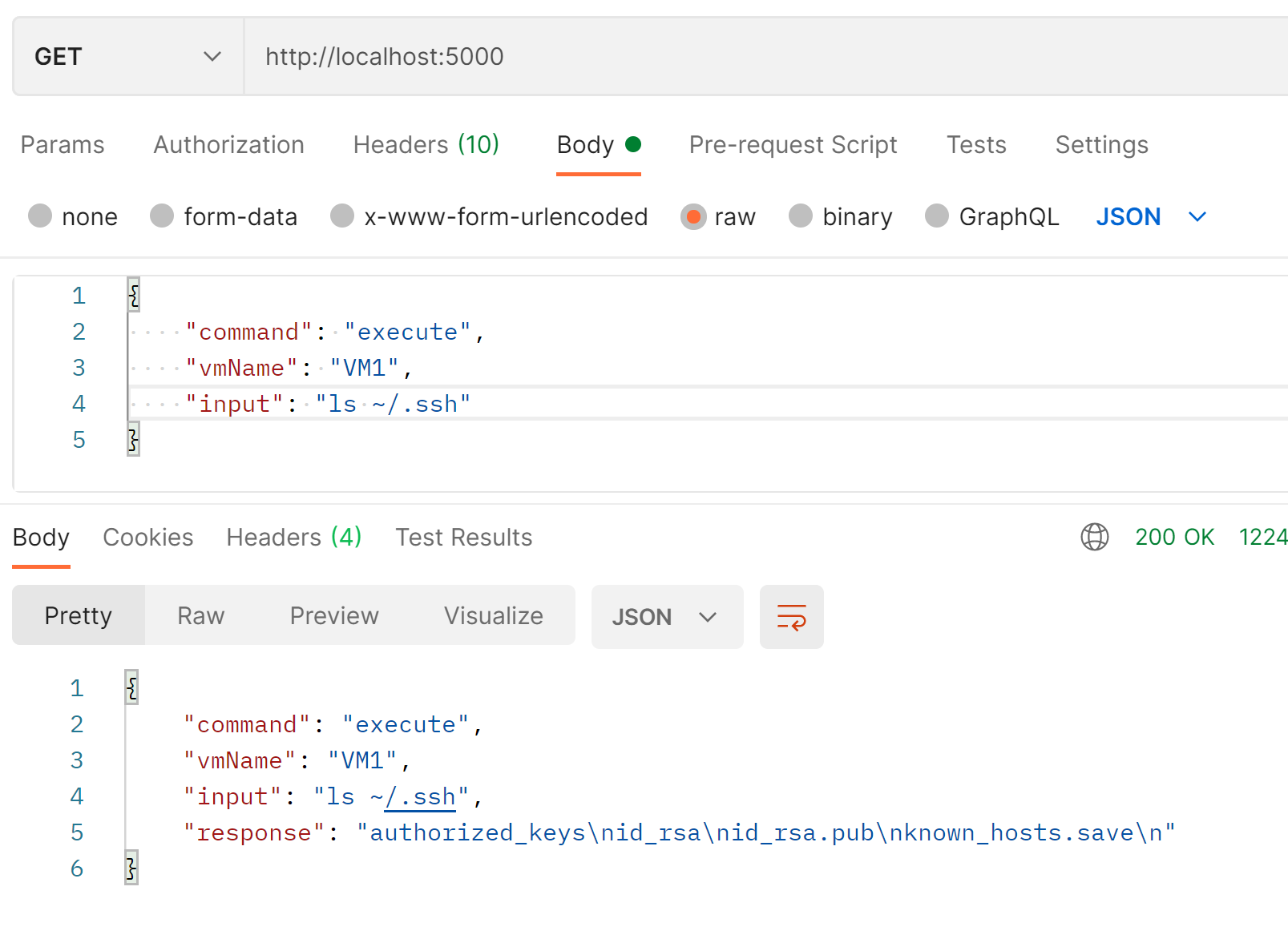
در این قسمت می‌توان مشخصات و تنظیمات این ماشین مجازی را تغییر داد:

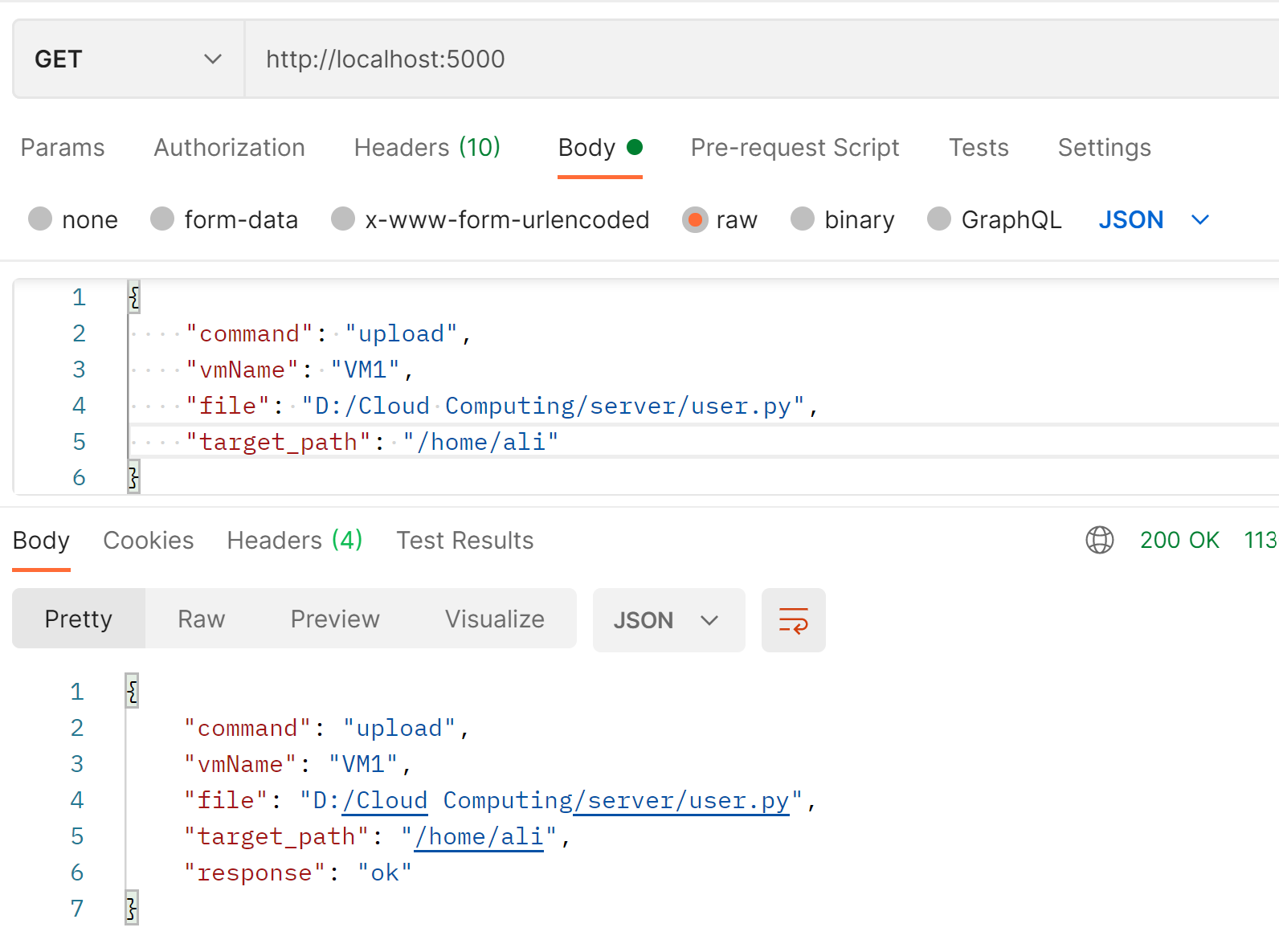


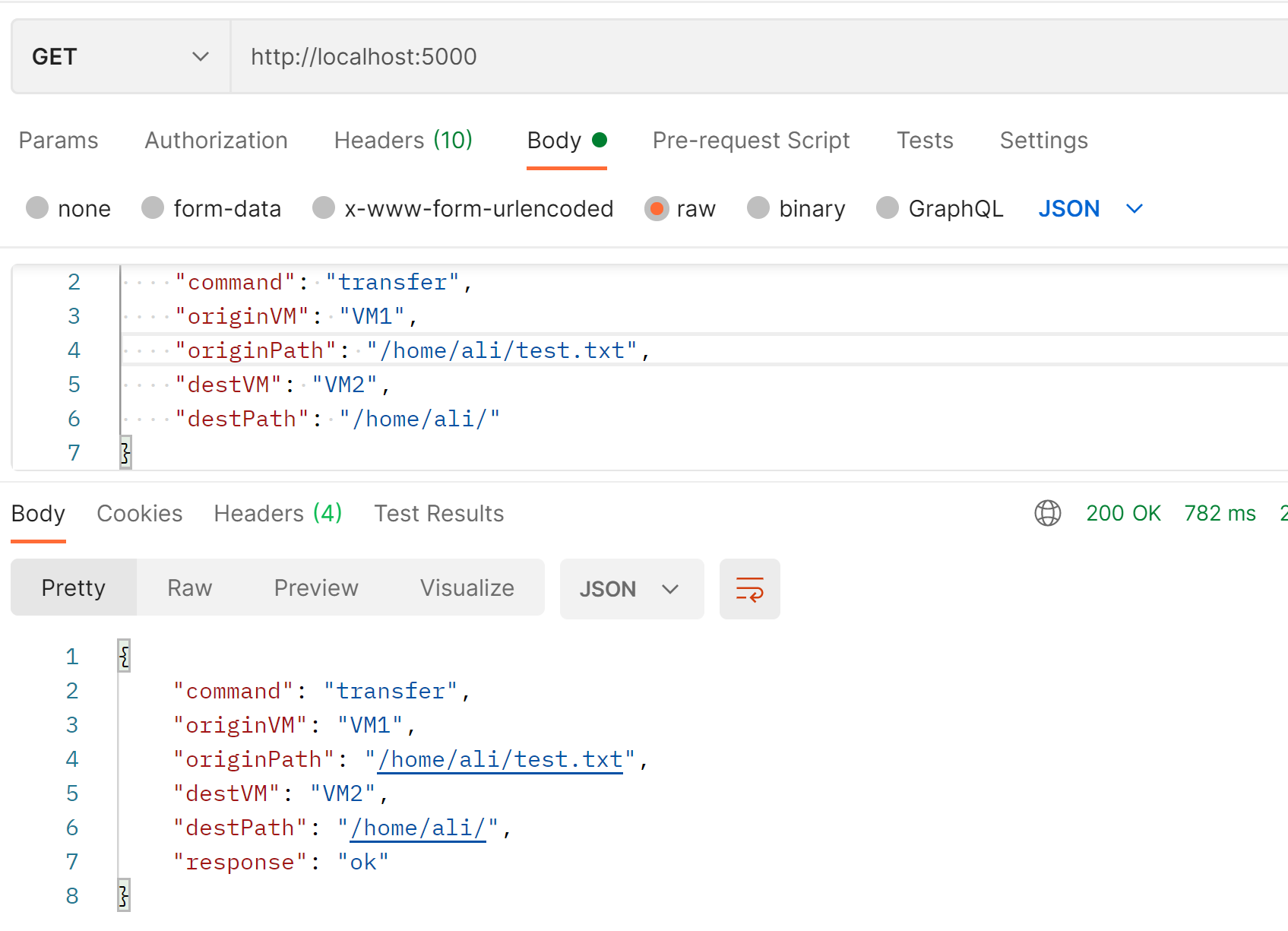
در این قسمت می‌توان یک ماشین را حذف کرد:



در این قسمت می‌خواهیم یک دستور را بر روی ماشین مجازی اجرا کنیم که برای آن ابتدا باید guest addition را بر روی ماشین فعال کنیم که با کمک لینک زیر این کار را انجام دادم:

<https://linuxize.com/post/how-to-install-virtualbox-guest-additions-in-ubuntu/>

در این قسمت می‌خواهیم یک فایل را بر روی یکی از ماشین‌ها آپلود کنیم:

و در قسمت‌ پایانی نیز بین ماشین‌ها فایلی را منتقل کردیم: