|  |
| --- |
|  |
| سید حمیدرضا میریگانه  متین برهانی |
| زمستان1402 |

# قسمت اول: کلیات OVS

## توضیح مختصر در مورد OVS:

Open vSwitch یا به اختصار OVS، یک پیاده‌سازی منبع باز از یک سوئیچ چند لایه مجازی توزیع شده است. [هدف اصلی Open vSwitch ارائه یک پشته سوئیچ برای محیط‌های مجازی‌سازی سخت‌افزاری است، در حالی که از چندین پروتکل و استاندارد مورد استفاده در شبکه‌های کامپیوتری پشتیبانی می‌کند](https://en.wikipedia.org/wiki/Open_vSwitch).

## نقاط قوت OVS:

1. پشتیبانی از استانداردها: OVS از چندین پروتکل و استاندارد مورد استفاده در شبکه‌های کامپیوتری پشتیبانی می‌کند.
2. قابلیت اتوماسیون شبکه: OVS قابلیت اتوماسیون شبکه را از طریق افزونه‌های برنامه‌نویسی فراهم می‌کند.
3. پشتیبانی از توزیع شفاف: OVS طوری طراحی شده است که پشتیبانی از توزیع شفاف در سرورهای فیزیکی متعدد را فراهم می‌کند.

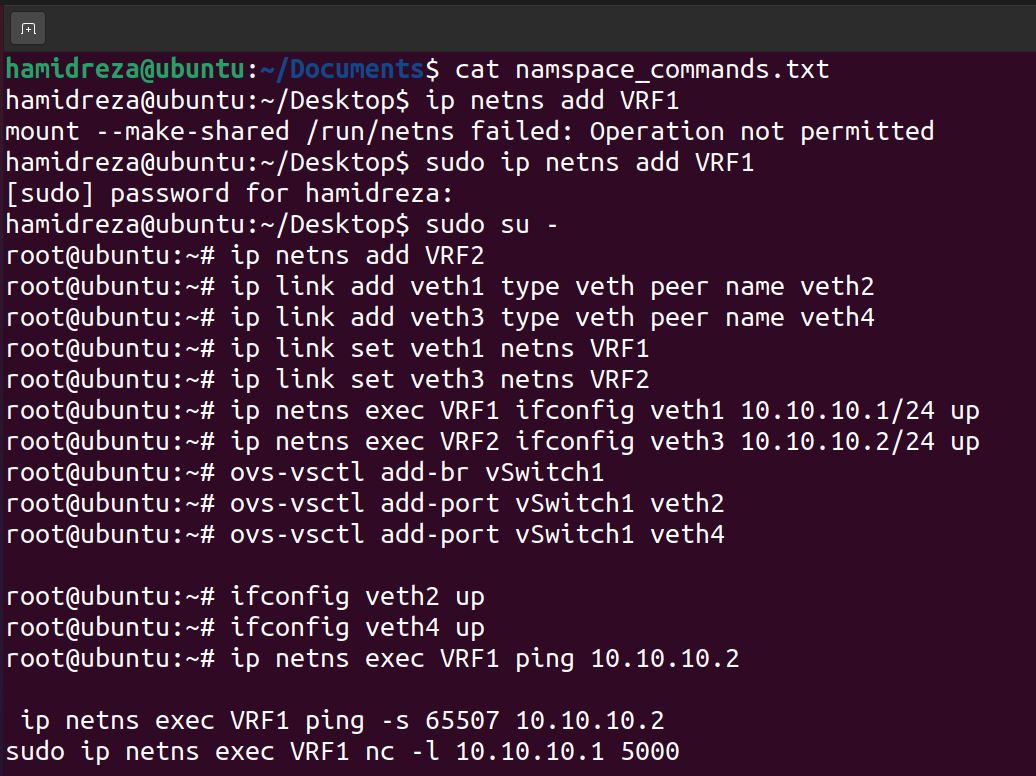
## نقاط ضعف OVS:

1. پیچیدگی در پیکربندی
2. نیاز به دانش فنی خاص برای استفاده بهینه

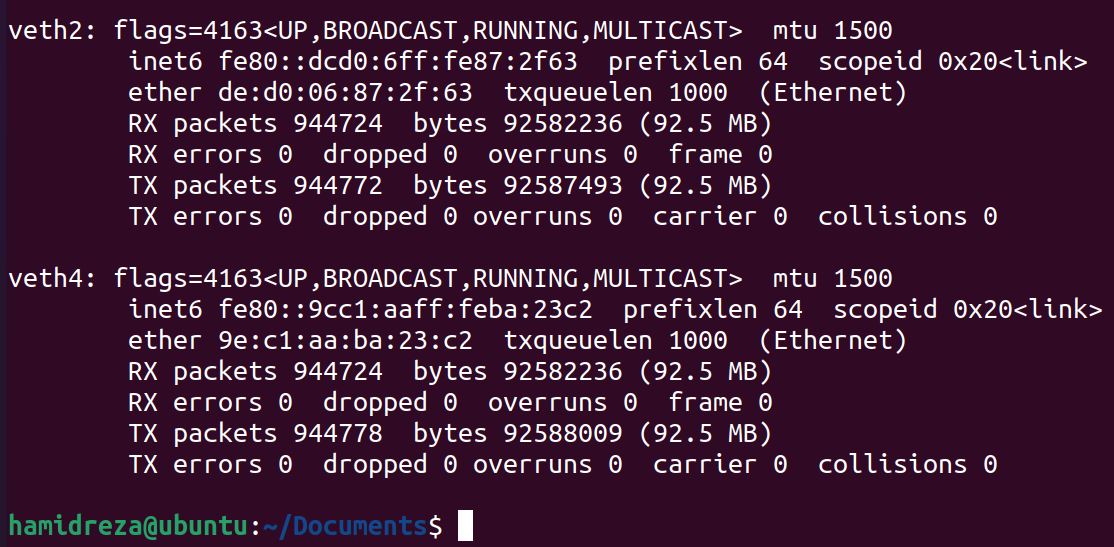
## پیاده سازی OVS:

در این قسمت به صورت مختصر، اقداماتی که در راستای ایجاد Switch مجازی شبکه انجام شد، شرح داده می‌شود.

1. در ابتدا با استفاده از امکانات لینوکس، اقدام به ایجاد دو Name Space به نام های VRF1 و VRF2 کردیم.
2. سپس اقدام به ایجاد دو Interface شبکه به نام های veth1 و veth2 و veth3 و veth4 کردیم.
3. سپس Interface با نام veth1 را به Namespace VRF1 و veth3 را به Namespace VRF3 اختصاص دادیم.
4. سپس به هر Interface یک IP منحصر به فرد در Range 10.10.10.255 اختصاص داده شد.
5. در ادامه با ایجاد یک Virtual Switch و ساخت Bridge بر روی آن، اقدام به برقراری ارتباط بین دو Namespace کردیم.



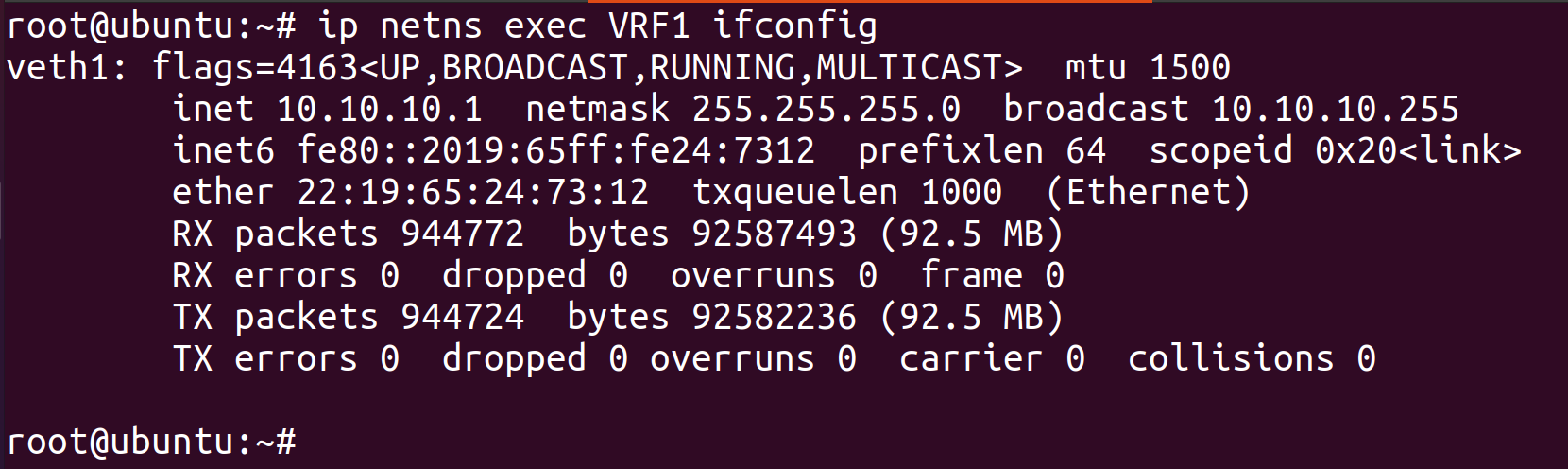
تصویر 1: دستورات ساختن Name Space و Interface های مربوطه



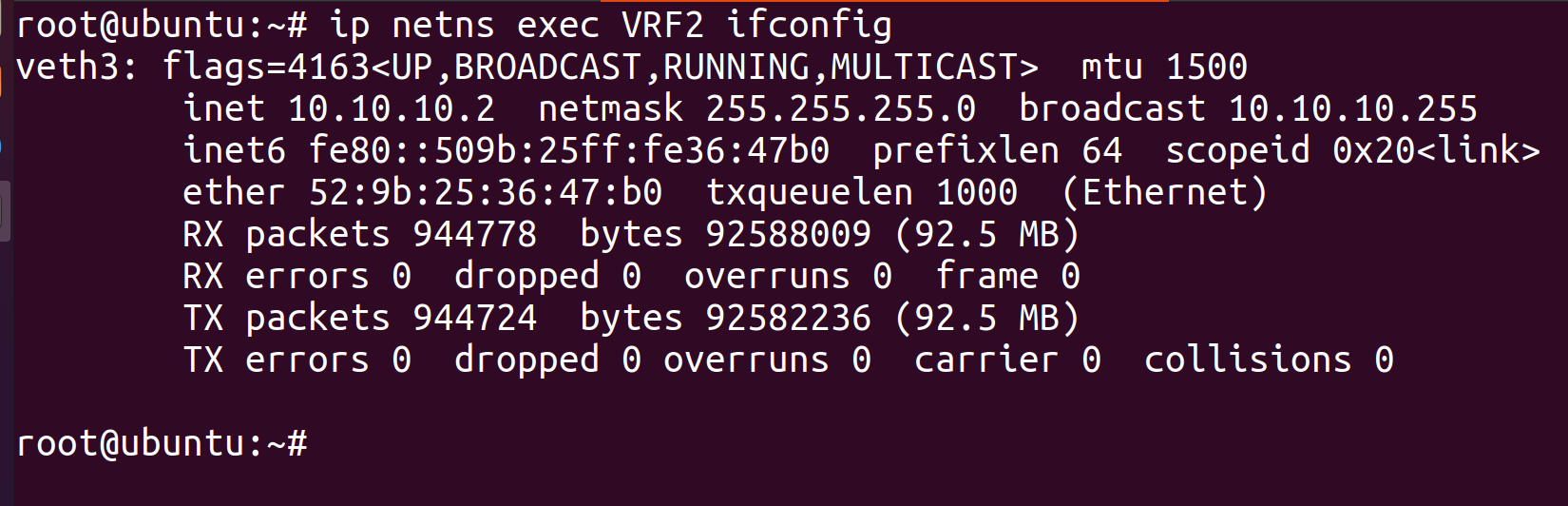
تصویر 2: Interface های ساخته شده



تصویر 3: مشاهده اطلاعات Switch مجازی تعریف شده



تصویر 4: Interface های Namespace به نامVRF1

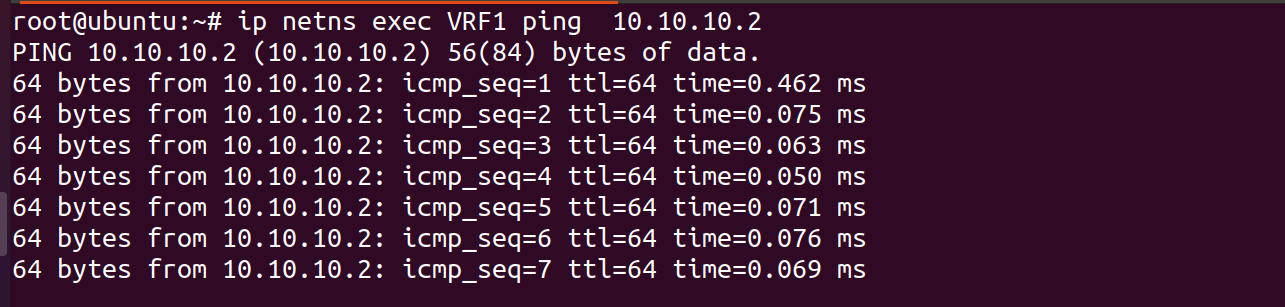


تصویر 5: ‏: ‏Interface‏ های ‏Namespace‏ به نامVRF2

# قسمت دوم: سناریوی Ping

## مقدمات سناریو Ping

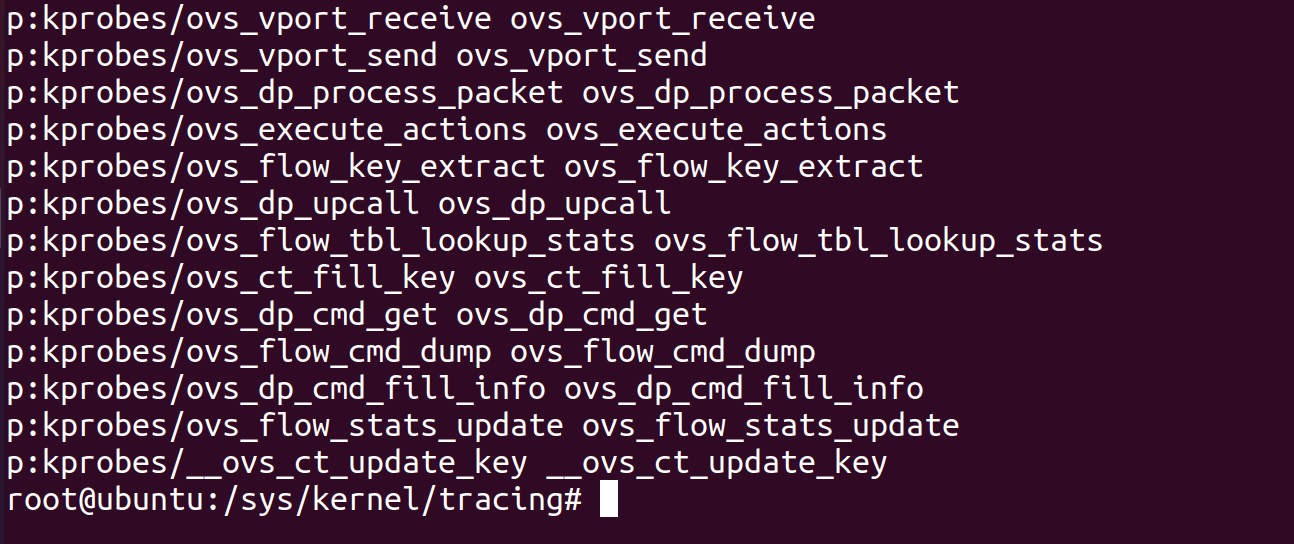
در ابتدا با استفاده از Name Space های ساخته شده، اقدام به ping کردن VRF1 و VRF2 کردیم. که در اشکال زیر مشهود است:



تصویر 6: ارسال دستور ICMP از طریق VRF1 به VRF2

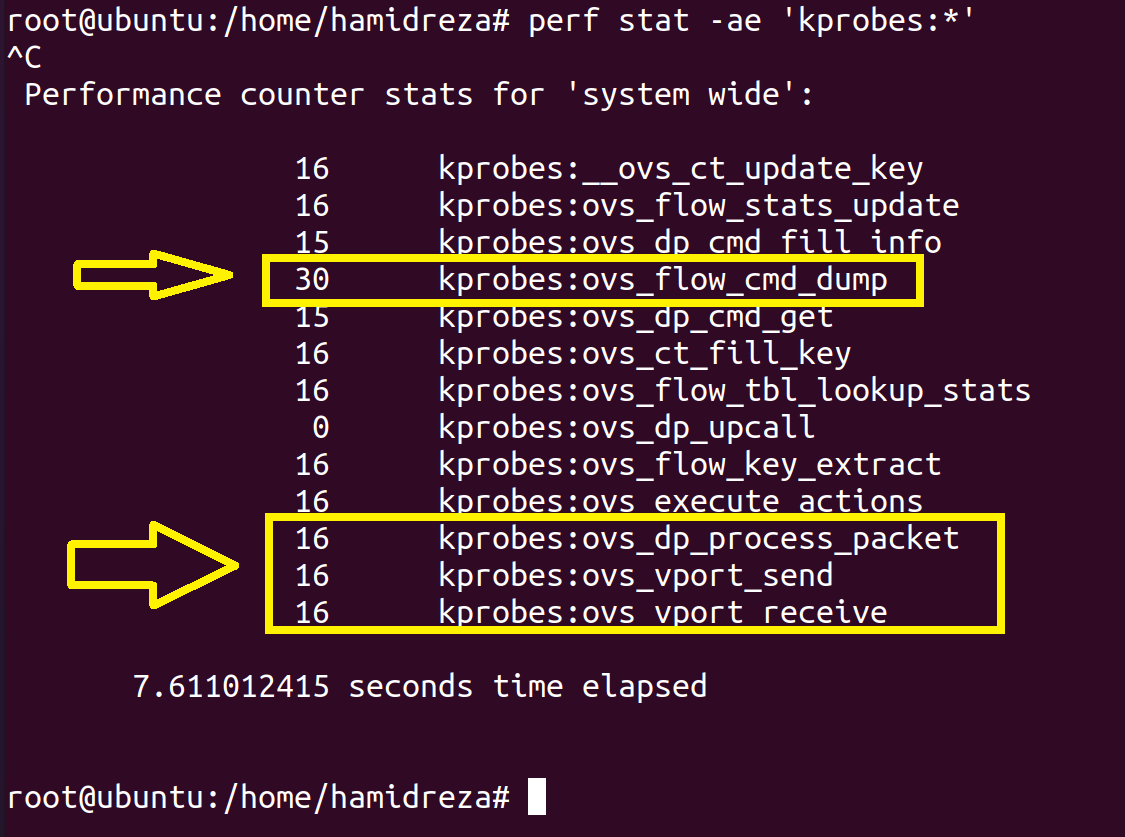
## دستور Perf

در ابتدا به علت نبود Event و یا Trace Point بر روی Kernel Module OVS، اقدام به تعریف کردن Kprobe بر روی Function های »رتبط با OVS کردیم. لیست مورد نظر به شرح زیر می باشد:



تصویر 7: لیست Kprobes های تعریف شده

سپس در ضمن اجرای دستور ping، اقدام به اجرای دستور perf stat بر روی این Kprobe ها نمودیم که ماحصل این دستور به شرح زیر می‌باشد:



تصویر 8: مشاهده اطلاعات آماری Event های صدا زده شده در دستور perf

پس از بررسیو آنالیز صورت گرفته، اقدام به بررسی عمیق تر Event های ذیل با استفاده از دستور نمودیم:

* ovs\_vport\_send
* ovs\_vport\_receive
* ovs\_dp\_process\_packet
* ovs\_flow\_cmd\_dump

## رویداد ovs\_vport\_send:

ابتدا اقدان به تهیه Call Stack با اتسفاده از دستور perf از این رویداد کردیم:

