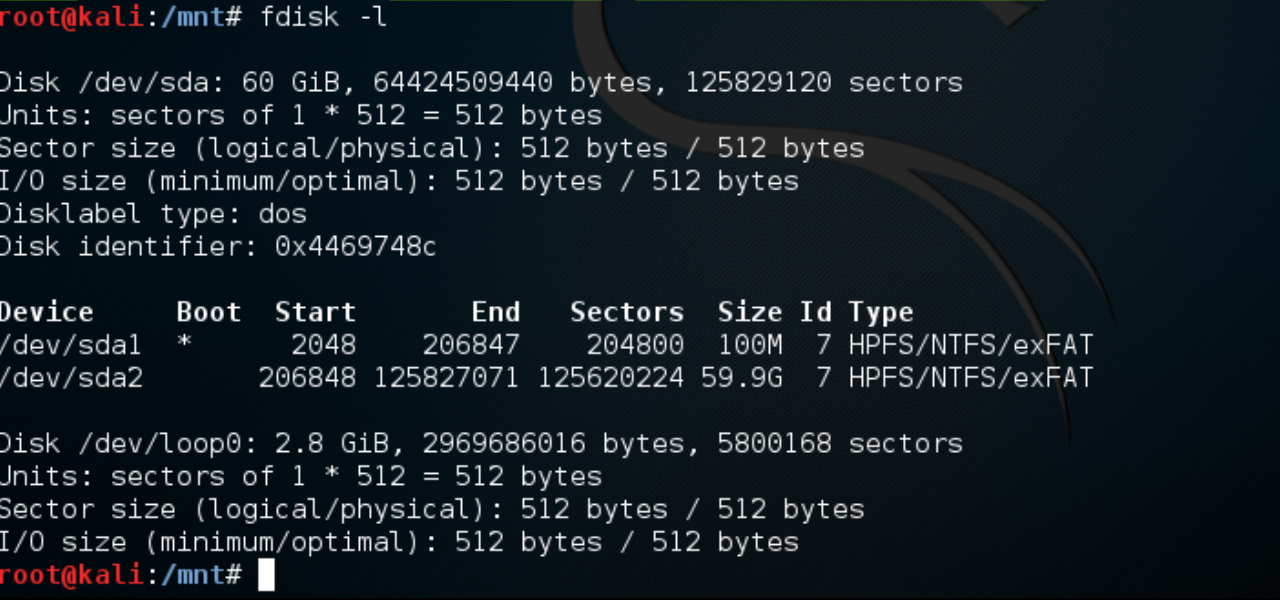
Assignment 3 Writeup

Step 1: Create Multiple accounts on windows machine and restart machine.

Step 2: Use kali linux iso file to launch kali at startup.

Step 3: Find mountable disks using command:

fdisk –l

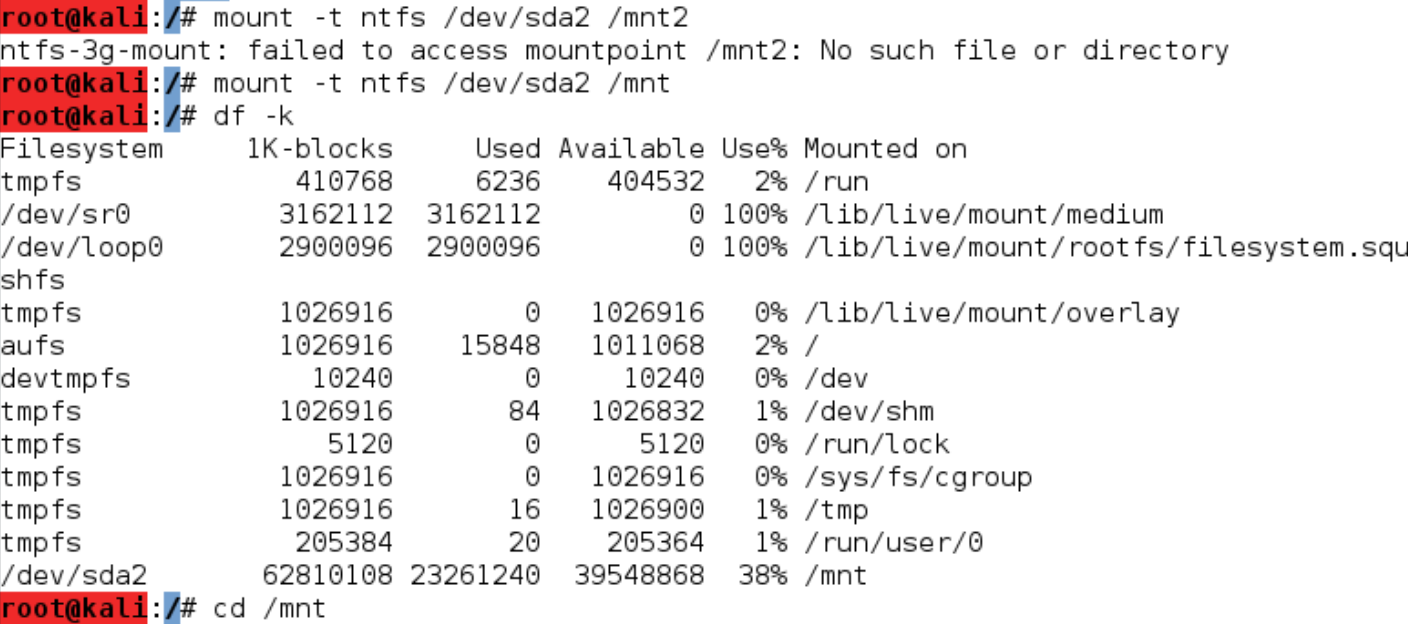


Step 4: Mount disk using command:

mount -t ntfs /dev/sda2 /mnt

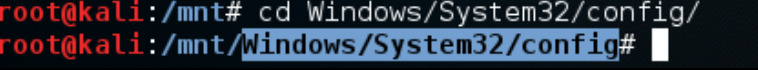
Step 5: Check the mounted disk(mount point) using command:

df –k



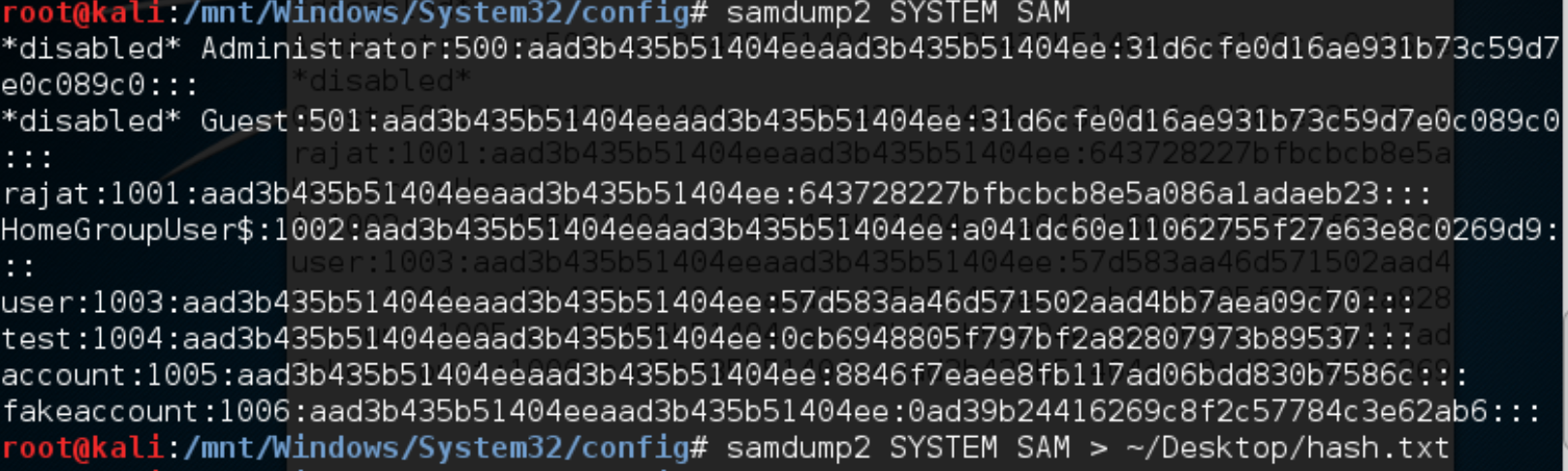
Step 6: Move to Windows/System32/config folder using:

cd Windows/System32/config



Step 7: Use samdump to get password hashes:

I have used pwdump to get the password hashes again as I was getting \*disabled\* for Admin and Guest account with samdump2 as shown below.



Step 8. Use John the Ripper to crack password on AWS EC2 instance:

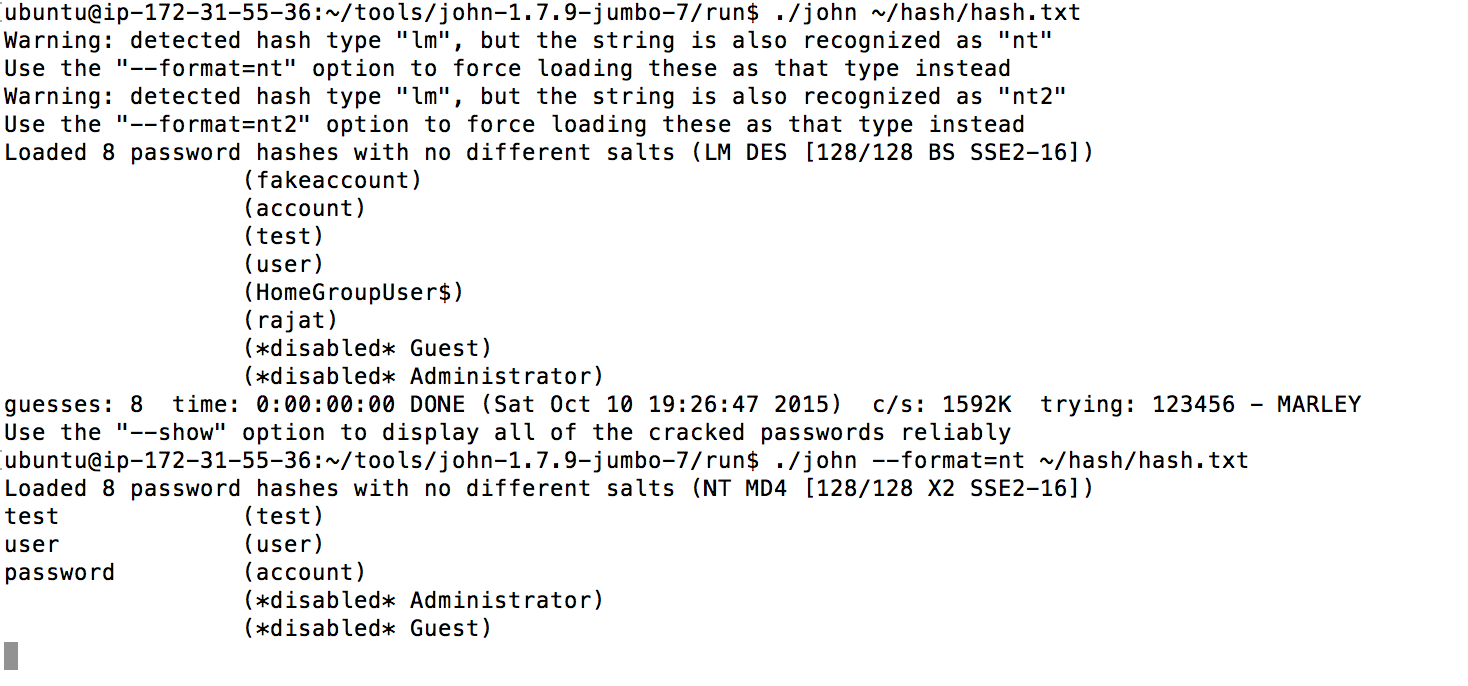
I install john using:

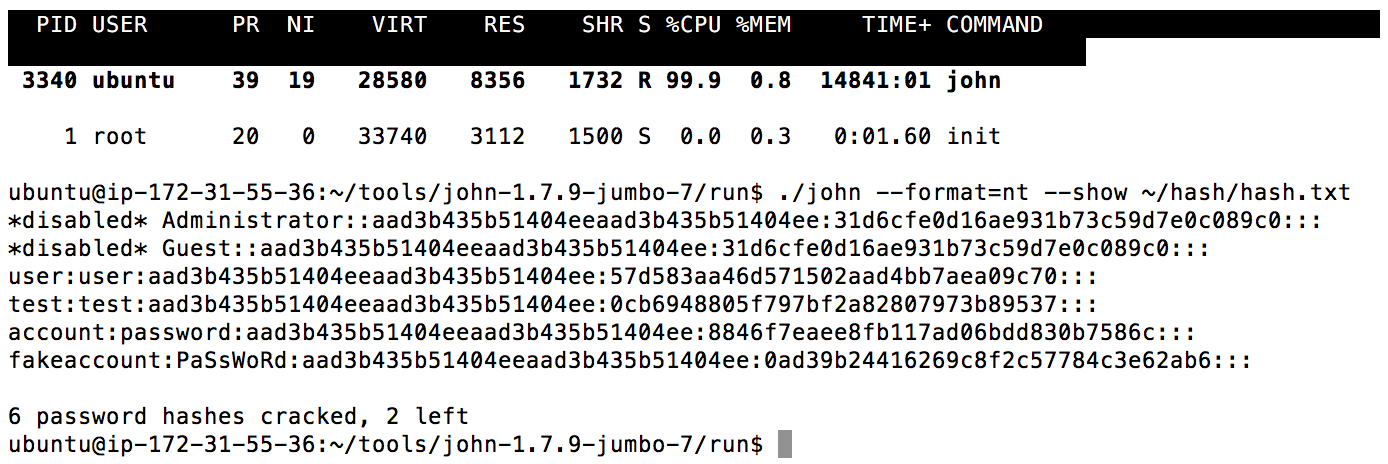
Sudo apt-get install john

But this version didn’t have NT format so it couldn’t crack my password file as it returned hashes itself assuming the hash format to be LM.

So I installed jumbo version of john using <http://samiux.blogspot.com/2013/05/howto-john-ripper-on-ubuntu-desktop.html> blog

After installing jumbo version of john:



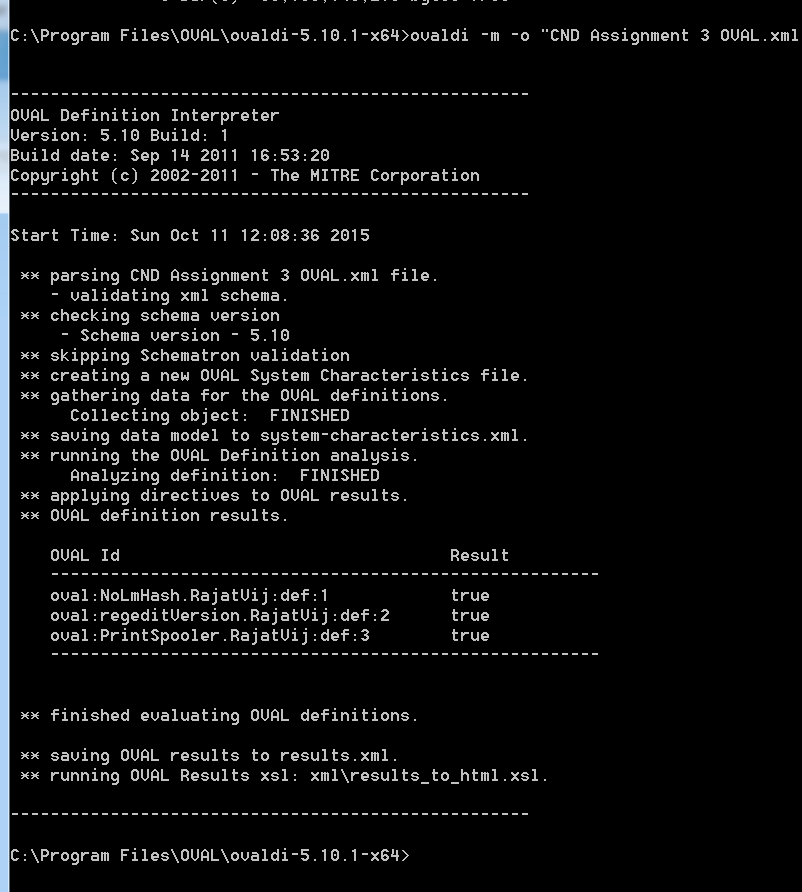
After 10 days:

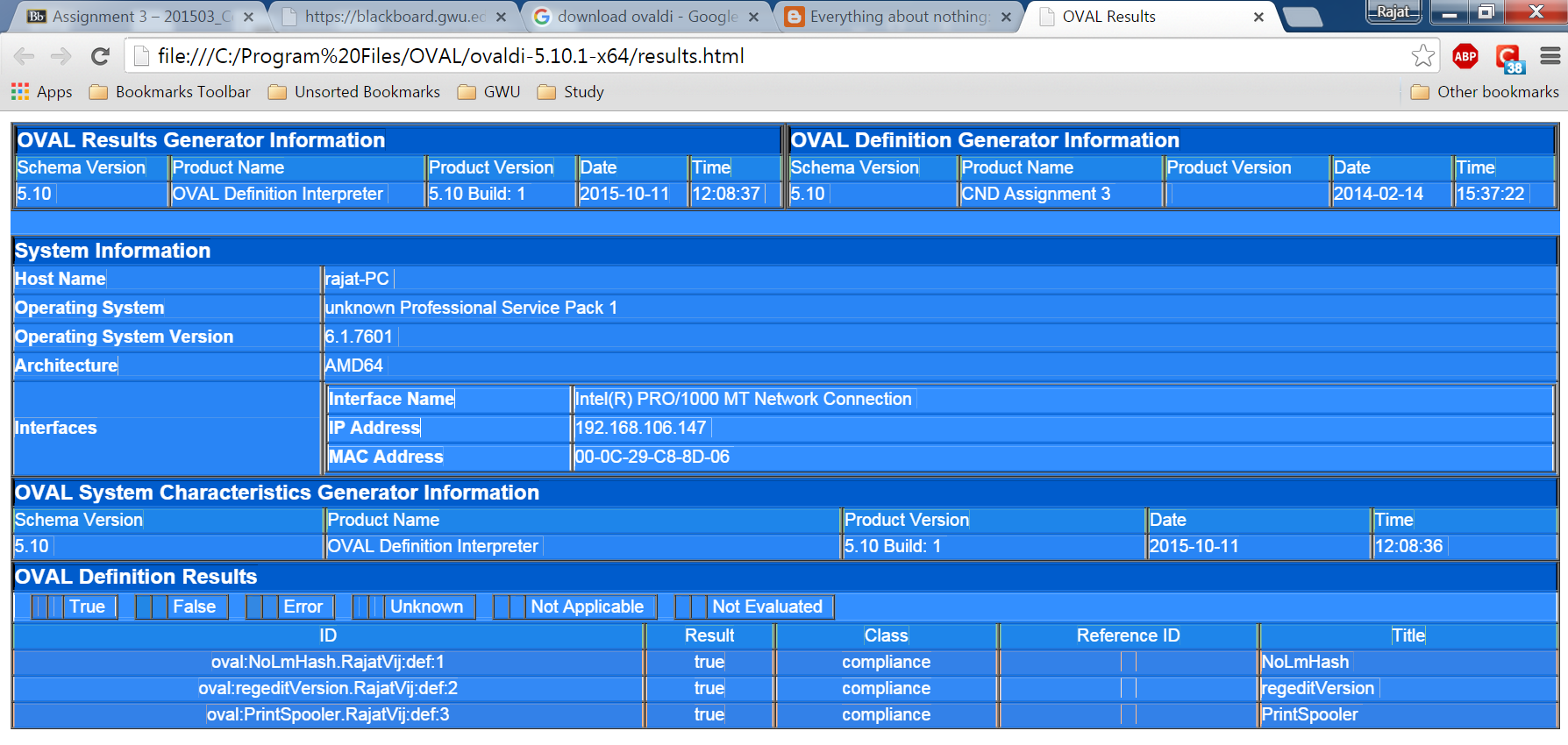
Ans. 2. OVAL Definations

Please refer to the following files in zip folder:

* CND Assignment 3 OVAL.xml
* Results.xml
* Results.html

Refered following link to create OVAL definition xml file. <https://oval.mitre.org/language/about/definition.html>



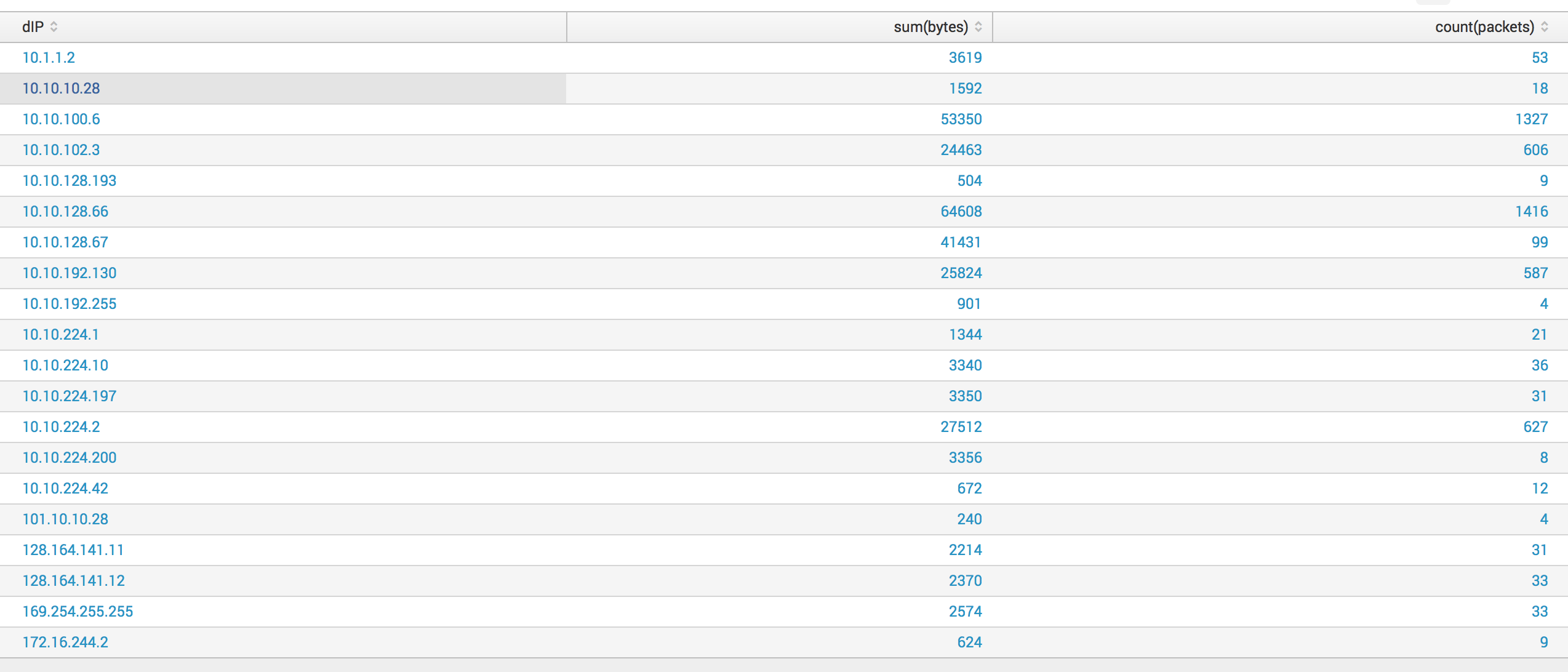
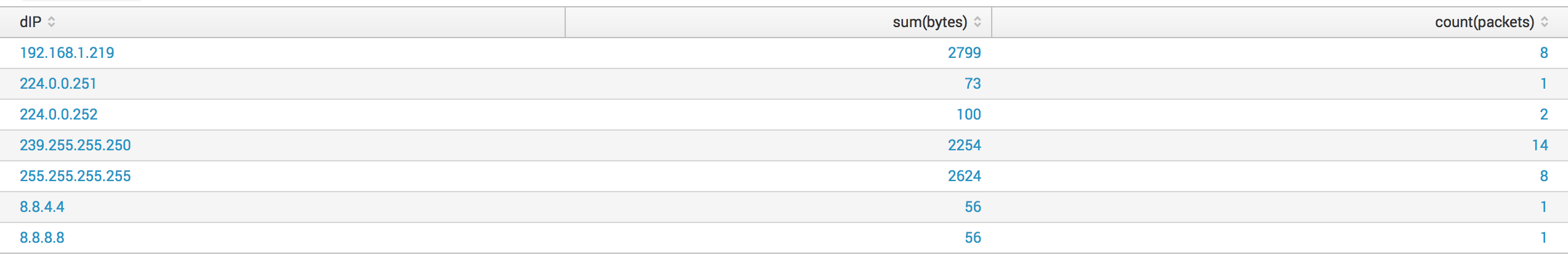


Refered following link to compile and get result html screen.

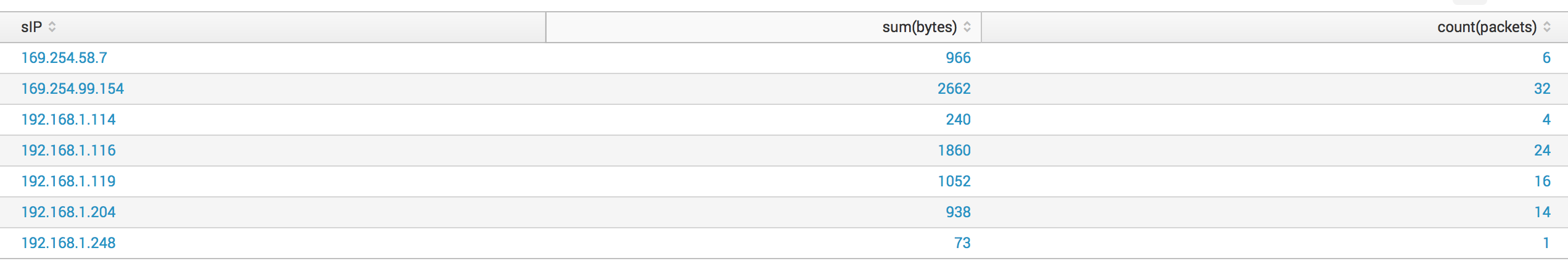
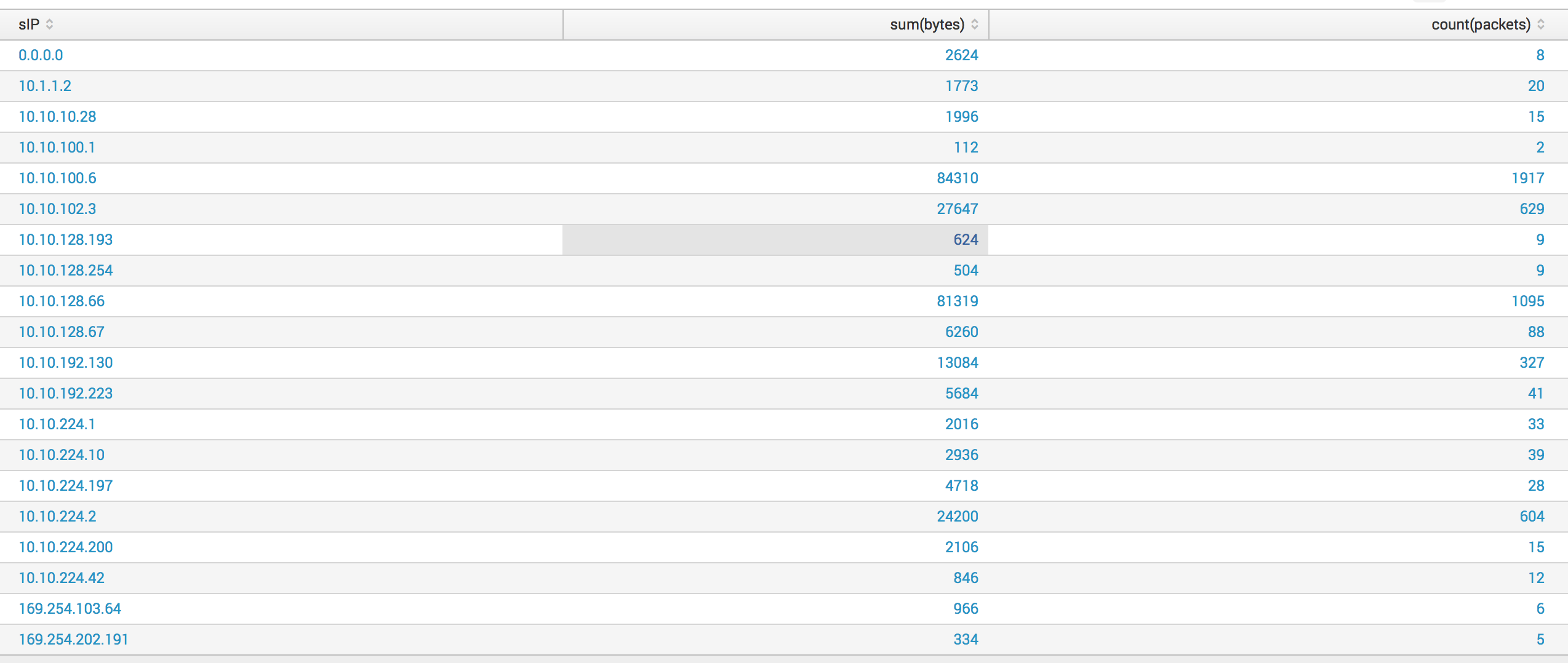
<http://sgros.blogspot.com/2011/10/installing-and-testing-ovaldi-on.html>

Ans. 3. Splunk

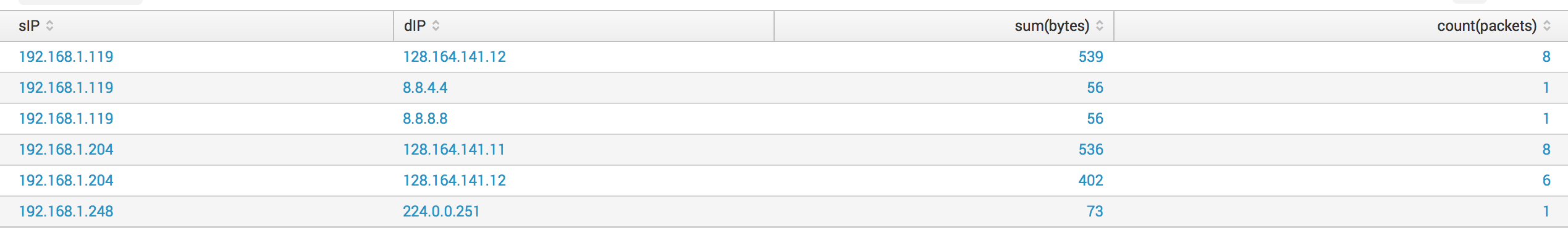
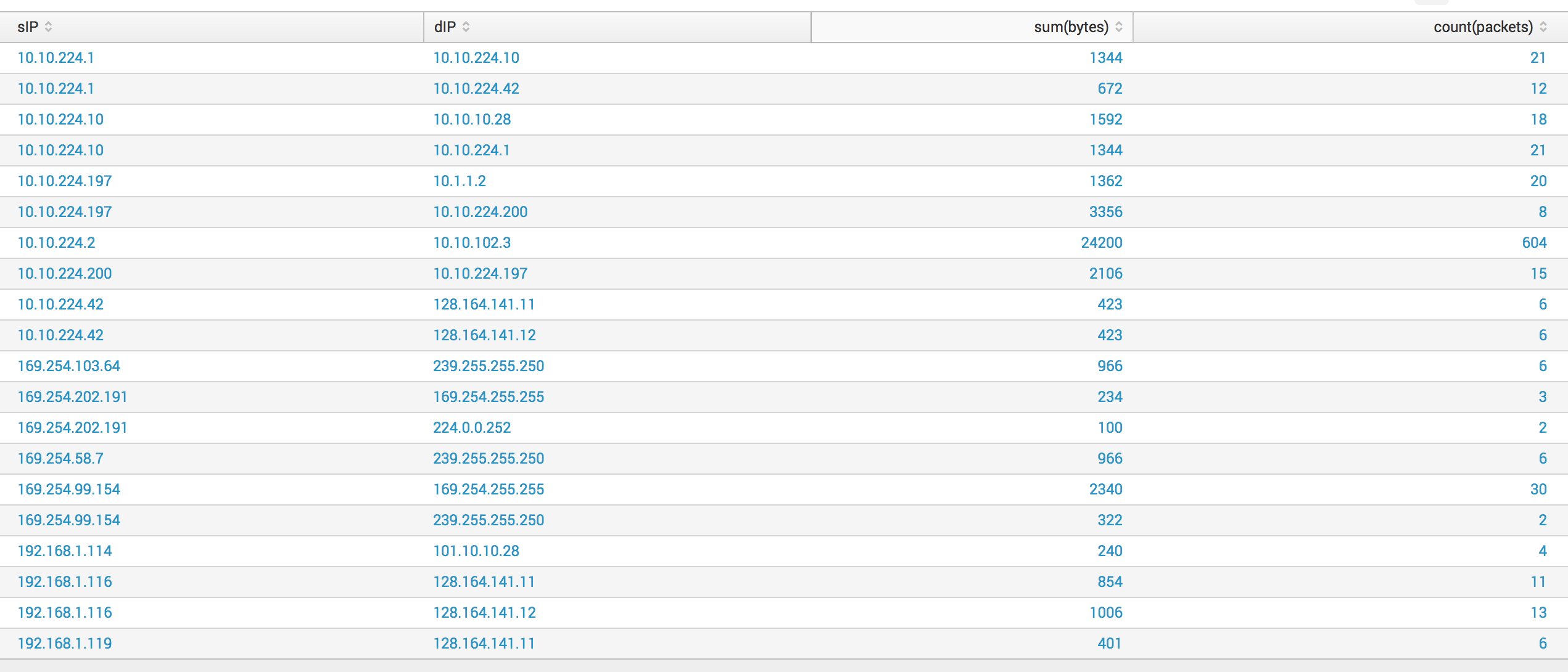
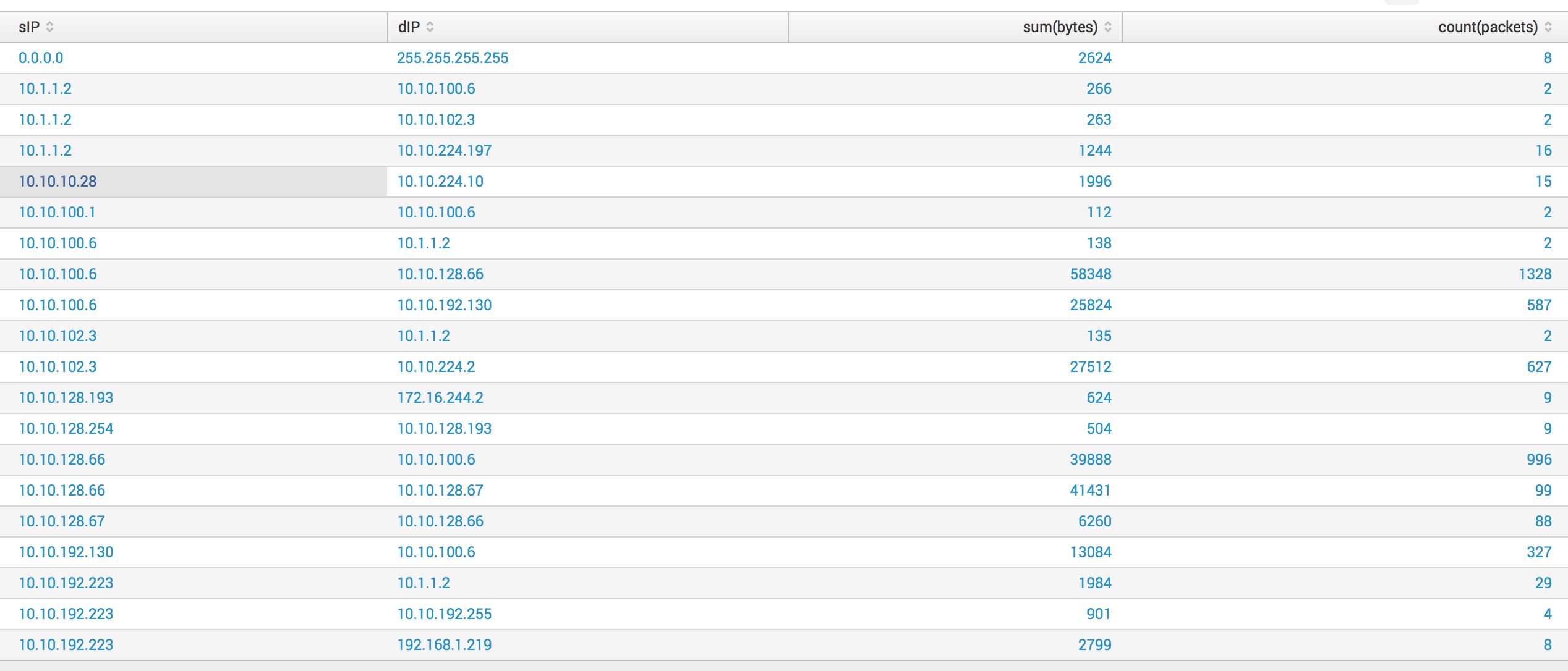
1. To determine sum of bytes, and the count for source IP and destination IP and their pair.
2. source="netflow\_example\_for\_class.csv" host="Rajats-MacBook-Pro.local" sourcetype="csv" | stats sum(bytes), count(packets) by dIP



1. source="netflow\_example\_for\_class.csv" host="Rajats-MacBook-Pro.local" sourcetype="csv" | stats sum(bytes), count(packets) by sIP



1. source="netflow\_example\_for\_class.csv" host="Rajats-MacBook-Pro.local" sourcetype="csv" | stats sum(bytes), count(packets) by sIP dIP



1. & 3. Regular expressions to get third and fourth octet in source IP and destination IP

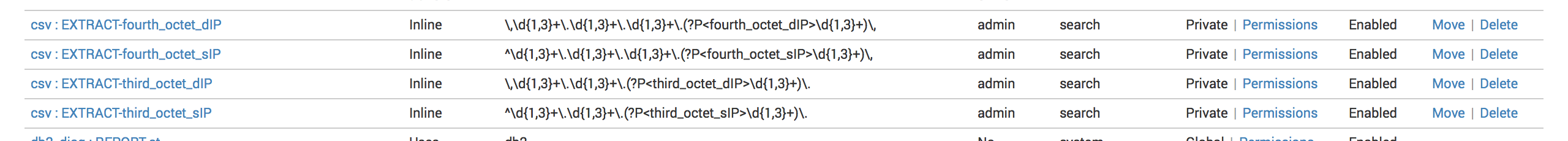
Regular expression:

Third Octet of Source IP = ^\d{1,3}+\.\ d{1,3}+\.(?P<third\_octet\_sIP>\d{1,3}+)\.

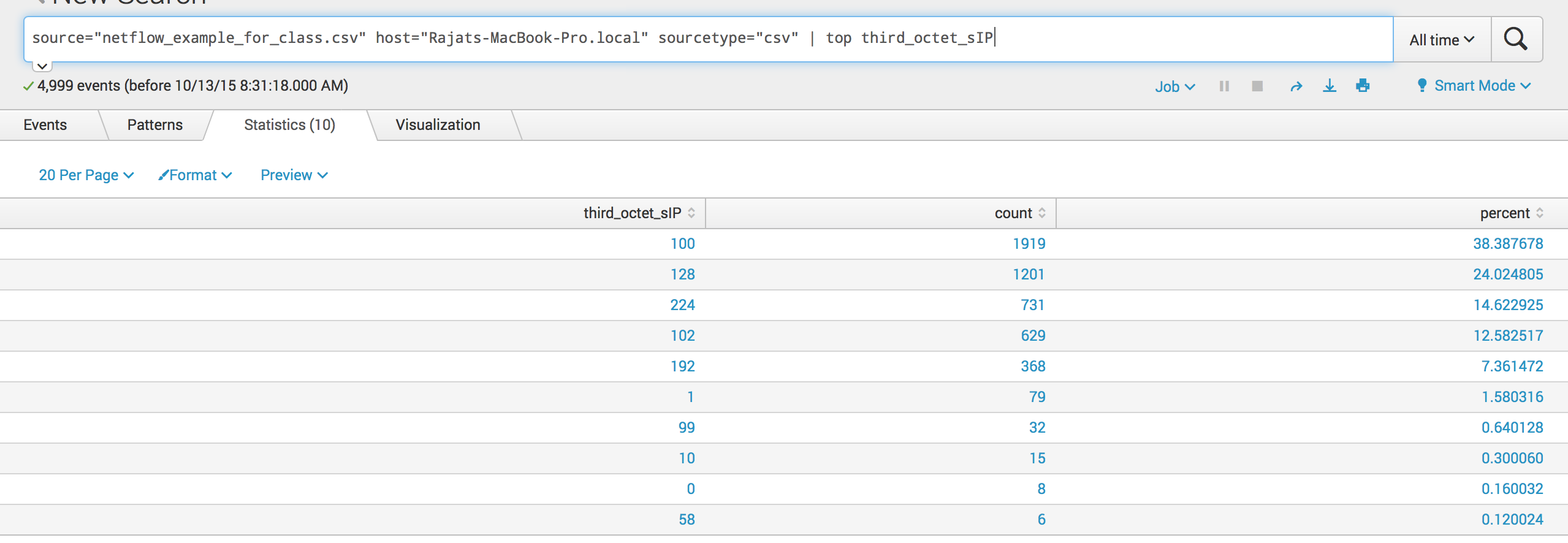
Third Octet of Destination IP = \,\d{1,3}+\.\ d{1,3}+\.(?P<third\_octet\_dIP>\d{1,3}+)\.

Fourth Octet of Source IP = ^\d{1,3}+\.\ d{1,3}+\.\ d{1,3}+\.(?P<fourth\_octet\_sIP>\d{1,3}+)\,

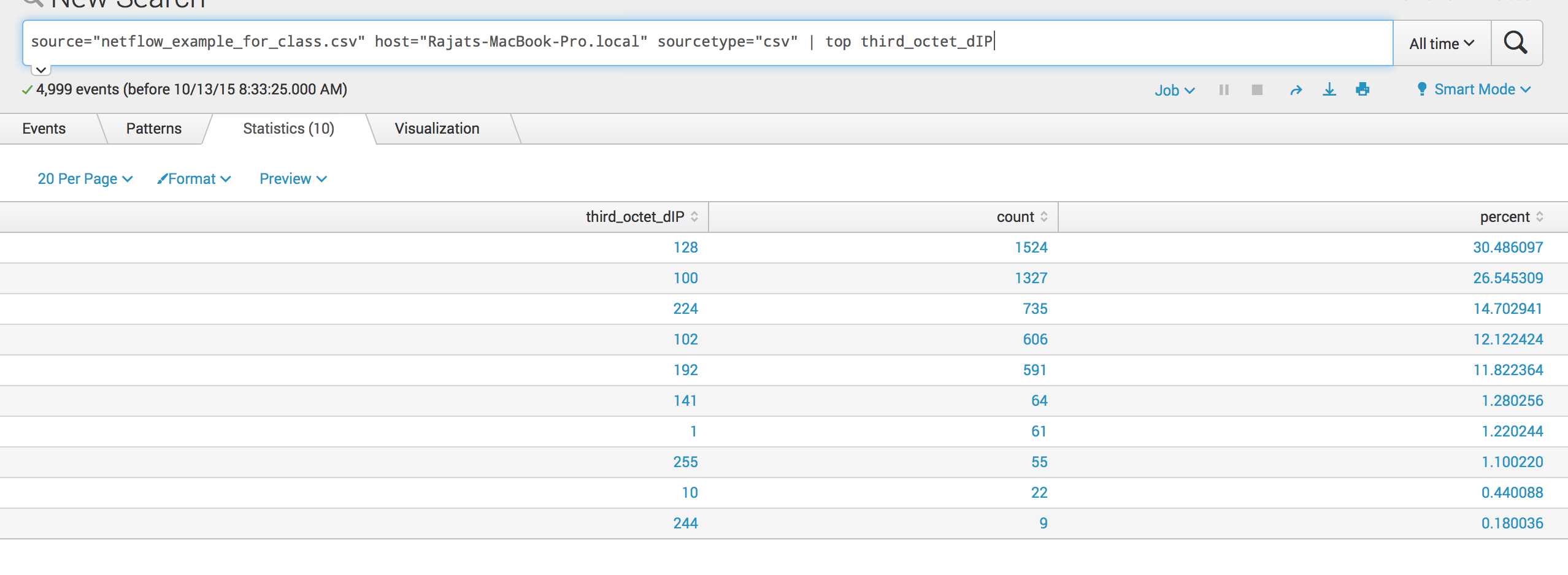
Fourth Octet of Destination IP =\,\d{1,3}+\.\d{1,3}+\.\d{1,3}+\.(?P<fourth\_octet\_dIP>\d{1,3}+)\,



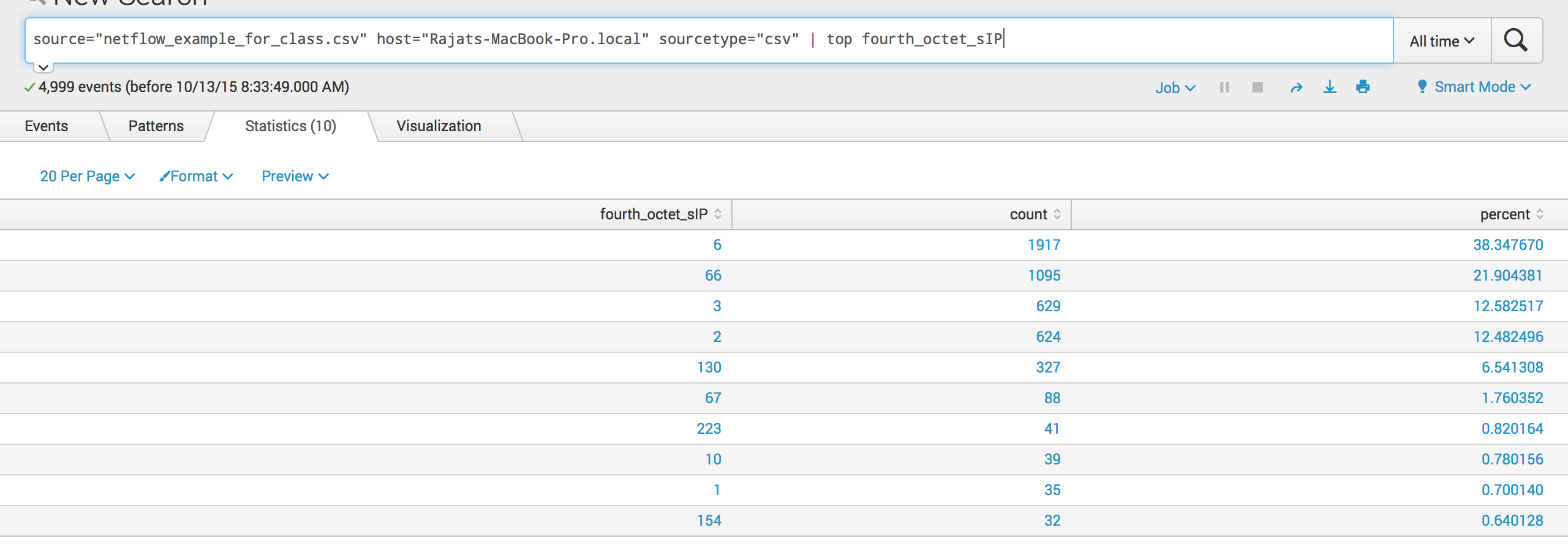
Most Common Third Octet of Source IP

Query : source=”netflow\_example\_for\_class.csv” host=”Rajats-Macbook-Pro.local” sourcetype=”csv” | top third\_octet\_sIP

Most Common Third Octet of Destination IP

Query : source=”netflow\_example\_for\_class.csv” host=”Rajats-Macbook-Pro.local” sourcetype=”csv” | top third\_octet\_dIP 

Most Common Fourth Octet of Source IP

Query : source=”netflow\_example\_for\_class.csv” host=”Rajats-Macbook-Pro.local” sourcetype=”csv” | top fourth\_octet\_sIP 

Most Common Fourth Octet of Destination IP

Query : source=”netflow\_example\_for\_class.csv” host=”Rajats-Macbook-Pro.local” sourcetype=”csv” | top fourth\_octet\_dIP 