

- 1) free muestra la cantidad total de memoria libre, usada y de swap en el sistema, así como los búferes y cachés utilizados por el núcleo.
- 2) El sistema tiene 7930 megas de memoria, de la cual 3734 megas están libres, pues teniendo en cuenta que al momento de hacer el análisis habían 275 procesos y Chrome abierto con 10 pestañas, me parece si son los valores que esperaba.
- 3) Código
- 4) La memoria cambia de acuerdo al uso que se le da, si se reserva un vector de 1024 posiciones y se recorre llenándolo con un valor inicial la memoria "free" se reducirá en una GB, la memoria usada aumenta la misma cantidad, la memoria disponible disminuye un poco. Al finalizar el proceso con el comando kill, el proceso finaliza y la memoria reservada es devuelta y sale de la memoria usada. Si coinciden con mis valores esperados, ya que después de ejecutar el comando kill la memoria reservada es liberada. Si reservo una cantidad de memoria superior a la "free" el sistema controla el posible error y no reserva nada.
- 5) La diferencia es que el comando free me entrega el total de la memoria, y las cantidades libres y usadas de la misma por todos los procesos, mientras que el comando pmap necesita de un/unos PID para mostrar el consumo de memoria de el o los procesos que se preguntaron en el PID.
- 6) Código
- 7) Si se ejecuta el pmap con -x nos encontramos con más detalles como el gasto de memoria, la aparición del RSS y el Dirty en kilobytes, con -d vemos que aparece el offset y el device y realizamos la última prueba con -q y es exactamente igual al pmap pero sin el total en kilobytes.
- 8) Al ejecutar pmap con diferentes cantidades de memoria nos encontramos que el total de la memoria usada es cada vez mayor conforme se le asigna más cantidad, es el resultado esperado teniendo en cuenta que a mayor asignación de memoria, el total de memoria usada por el proceso deberá ser mayor.