The following data was gathered using the vm\_pageout\_scan function. We used counter variables to count all the pages in our data and each variable is explained in the design document. The command we used to stress was: stress –vm- 17 –timeout 60s –vm-keep.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Scan # | # of pages in inactive queue scanned | # of pages in active queue scanned | # of pages in inactive queue | # of pages in active queue | # of pages scan | # of pages queue for flush | # of pages move from inactive to cache/free | # of pages from inactive to active | # of pages from active to inactive |
| 1 | 0 | 40327 | 4127 | 402879 | 40327 | 0 | 0 | 0 | 0 |
| 2 | 4127 | 973854 | 763 | 976825 | 977981 | 0 | 2067 | 1321 | 0 |
| 3 | 763 | 976827 | 856 | 976588 | 977590 | 726 | 146 | 0 | 0 |
| 4 | 856 | 171602 | 49225 | 927373 | 172458 | 0 | 854 | 2 | 37 |
| 5 | 49446 | 274703 | 94841 | 881568 | 324149 | 49334 | 189 | 0 | 221 |
| 6 | 58352 | 111805 | 82965 | 883162 | 170157 | 4 | 13319 | 0 | 0 |
| 7 | 17709 | 6064 | 44871 | 918872 | 23773 | 0 | 17697 | 12 | 0 |
| 8 | 34154 | 163659 | 52534 | 910694 | 197813 | 17543 | 16617 | 45 | 0 |
| 9 | 34405 | 174609 | 59055 | 906974 | 209014 | 21199 | 13185 | 62 | 48 |
| 10 | 27542 | 59107 | 52678 | 913732 | 86649 | 13826 | 13779 | 42 | 144 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | # of pages in inactive queue scanned | # of pages in active queue scanned | # of pages in inactive queue | # of pages in active queue | # of pages scan | # of pages queue for flush | # of pages move from inactive to cache/free | # of pages from inactive to active | # of pages from active to inactive |
| 1 | 0 | 8 | 3015 | 4903 | 8 | 0 | 0 | 0 | 0 |
| 2 | 3047 | 974793 | 277 | 974793 | 977840 | 0 | 1567 | 1227 | 0 |
| 3 | 277 | 977377 | 451 | 977143 | 977654 | 240 | 65 | 0 | 5 |
| 4 | 451 | 977143 | 390 | 976785 | 977594 | 24 | 427 | 0 | 33 |
| 5 | 390 | 976785 | 614 | 976310 | 977175 | 139 | 251 | 0 | 0 |
| 6 | 614 | 111687 | 49167 | 927416 | 112301 | 311 | 341 | 0 | 0 |
| 7 | 49167 | 121489 | 59421 | 916384 | 170656 | 12382 | 778 | 36240 | 0 |
| 8 | 59421 | 86103 | 46370 | 927561 | 145524 | 0 | 12149 | 47272 | 0 |
| 9 | 46370 | 77003 | 70134 | 907476 | 123373 | 21420 | 17 | 24950 | 0 |
| 10 | 18119 | 782 | 52015 | 911822 | 18901 | 0 | 17696 | 423 | 0 |

We took the average of each column of each table and compared them:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
|  | # of pages in inactive queue scanned | # of pages in active queue scanned | # of pages in inactive queue | # of pages in active queue | # of pages scan | # of pages queue for flush | # of pages move from inactive to cache/free | # of pages from inactive to active | # of pages from active to inactive |
| Modified Kernel | 22735.4 | 295255.7 | 44191.5 | 869866.7 | 317991.1 | 10263.2 | 7785.3 | 148.4 | 45 |
|  |  |  |  |  |  |  |  |  |  |
| Stock Kernel | 17785.6 | 430317 | 28185.4 | 850059.3 | 448102.6 | 3451.6 | 3329.1 | 11011.2 | 3.8 |

The performance of the modified kernel closely resembles the stock kernel, which is expected because no big changes to the logic or algorithms were made, only the insertion places. We observe that our modified kernel moved pages to the inactive queue more quickly than the stock kernel did; however, once the memory was fully utilized, both kernels paged and kept roughly the same number of pages in each of the queues. The only big discrepancy shown in the data is the value for the pages scanned in the active queue; the modified kernel seemed to scan more pages on average.