**Linux capacity planning: 5 things you need to do**

Capacity planning is the "black box" of Linux system administration. Here are five things you need to do to get started.

Posted: September 23, 2019 | **by**[Ken Hess (Red Hat)](https://www.redhat.com/sysadmin/users/khess)

I think a lot of system administrators either fear capacity planning or just think it's unnecessary. First, there's no reason to fear capacity planning (it isn't rocket science); and second, capacity planning is 100% necessary. In the past, system administrators had to deal with management making sweeping decisions to add capacity and enhance performance, either by throwing new systems into the mix or adding CPU, RAM, or faster storage. Usually, but not always, the problem persisted beyond the upgrades and added capacity. But the "usually" qualifier is the part of the equation that stumps system administrators and managers alike—to the point that no one wants to deal with actual capacity and performance planning and administration.

This issue doesn't have to be a struggle. In this article, I present five things you need to know to get started in Linux capacity planning. You can also apply these guidelines to any environment: Linux, Windows, Unix, or a hybrid version of these.

Capacity planning basics

More Linux resources

* [Download Now: Linux Commands Cheat Sheet](https://developers.redhat.com/cheat-sheets/linux-commands-cheat-sheet/?intcmp=701f20000012ngPAAQ)
* [Advanced Linux Commands Cheat Sheet for Developers](https://developers.redhat.com/cheat-sheets/advanced-linux-commands/?intcmp=701f20000012ngPAAQ)
* [Download Red Hat Enterprise Linux Server 8 Trial](https://www.redhat.com/en/technologies/linux-platforms/enterprise-linux/try-it?intcmp=701f20000012ngPAAQ)
* [Linux System Administration Skills Assessment](https://www.redhat.com/rhtapps/assessment/session/?intcmp=701f20000012ngPAAQ)

When you discuss capacity, you're really discussing performance. Capacity and performance are always mentioned together. You have to measure and monitor performance to do any sort of capacity planning.

Capacity means the ability to process and store data without bottlenecks or impact to the end user. Most of the time, system administrators think of performance in terms of data processing for websites, databases, or applications. But, that's not where performance ends. Think about backup and restore performance. Backups require compression, deduplication, disk-to-disk transfer, or over-the-network transfer. And don't forget, moving virtual machines from one host to another requires compute, storage, AND network capacity.

First: Get a baseline

It doesn't matter whether your systems are brand new or three years old, you must establish a baseline before you can begin capacity planning and projection. Establishing a baseline is somewhat time-consuming because a baseline is not a snapshot, it is rather a longer-term view of performance. Use at least a one-month baseline for each system. A month of data should give you the range of performance from which you can plan and forecast capacity needs.

There are three numbers you need to examine after you've gathered the preliminary date: peak, low, and average load or usage. After analyzing this data, you'll realize why you cannot depend on a load snapshot to guide you through the capacity planning process. A baseline tells you where you are in this process.

The next set of data you need to consider is current capacity. You need to assess RAM, CPU, disk, and network capacity information. Then, you need to find out what your maximum capacity is for each system. The difference between current and maximum capacity gives you your growth capacity. For example, consider a system that has the following configuration: two Quad-core CPUs, 128GB RAM, two 1TB disks in RAID 1 (mirrored), and one dual Gb Ethernet network interface. Your maximum capacity for this system, then, is four quad-core CPUs, 512GB RAM, six disks, and two open PCIe slots for expansion cards such as Gb Ethernet network interface cards (NICs).

|  | | |
| --- | --- | --- |
|  | **Current Capacity** | **Maximum Capacity** |
| **CPU** | 2 - Quad core | 4 - Quad core |
| **RAM** | 128 GB | 512 GB |
| **Disk** | 2 Disks - 1 TB - RAID 1 | 6 Disks |
| **NIC** | 2 GbE (Dual) | 6 GbE (Dual) - 10 GbE (Quad) |

Now, compare the two. This system has much more available capacity for increased compute power, network, and storage. These hardware capacity parameters plus your month's worth of performance data are your starting points in predicting the need for extra capacity, whether in the form of system upgrades or a complete technology refresh.

Second: Set up performance monitoring

If you don't already have a performance monitoring package such as sysstat installed, you can easily do so from the default repositories. Check to see if you have sysstat:

**$** rpm -qa |grep sysstat

If you don't have it, then install it with:

**$** sudo yum -y install sysstat

Execute the following two commands to run sysstat's data collectors on startup and then to start sysstat's data collectors on your system:

**$** sudo systemctl enable sysstat sysstat-collect.timer sysstat-summary.timer

**$** sudo systemctl start sysstat sysstat-collect.timer sysstat-summary.timer

The sysstat package consists of a handful of commands that report performance statistics on a variety of subsystems and services from CIFS/Samba to disk to Linux tasks. The most useful command is sar, the System Activity Reporter. The sar command provides you with a running list of system activity statistics. Any user can issue the sar command to view statistics:

**$** sar

Linux 4.18.0-80.7.1.el8\_0.x86\_64 (rhel) 08/14/2019 \_x86\_64\_ (1 CPU)

12:00:24 AM CPU %user %nice %system %iowait %steal %idle

12:10:01 AM all 0.22 0.00 0.43 0.01 0.00 99.33

12:20:32 AM all 1.18 0.05 1.24 0.12 0.00 97.41

12:30:01 AM all 0.27 0.00 0.49 0.01 0.00 99.23

12:40:32 AM all 0.20 0.00 0.38 0.00 0.00 99.41

12:50:32 AM all 0.18 0.00 0.36 0.01 0.00 99.46

By default, system statistics are gathered every 10 minutes. The sar command displays general system statistics, but the far more useful and extensive statistics option provides everything sar has to offer, using the -A option:

**$** sar -A

The output is far too long to post here, but note that you'll see all statistics for every subsystem and service that sar collects. Please refer to the sar man page for further information and details about specific statistics and their options.

Third: Analyze and plot data

The sysstat collector gathers system information and keeps it under /var/log/sa. The file numbers are the day of the month from which they were collected. You'll need some method of gathering and analyzing this data. I suggest Blair Zajac's [Orca](http://orcaware.com/orca). I also suggest that you transfer your collected data to a central repository to process and display. In other words, do not process your statistics on your production systems because it will negatively impact your performance statistics and skew your results.

Orca is non-trivial but not terribly difficult to set up. A few years ago, I wrote an [article](https://nnc3.com/mags/LM10/Magazine/Archive/2006/65/054-057_orca/article.html) that helps you start displaying your performance statistics on a web server with Orca. Orca hasn't been updated in a while, but still works as shown in the documentation and in my article.

Fourth: Set performance thresholds

For each of your production or monitored systems, you have to answer the question: "How busy is busy?" There is no perfect answer, and you will likely tweak the numbers at some point to reduce how many notifications you receive from breaching those thresholds. For example, let's say that you have five web servers that are load-balanced to supply web services to your external customers, and you need to monitor their performance to predict when to add more systems to the farm, or when you can take one or more offline.

As a preliminary test, you set the CPU threshold at 80% busy for all five servers. Twice per day, you receive email alerts that your systems have gone above 80%. The problem? You receive alerts every five minutes from all five servers, for two hours, twice per day. This is an indication that the thresholds are set too low unless you love getting all those notifications.

You have to look at performance during those peak times to decide where to set the threshold, and whether or not you need to add more systems to the farm to lower the overall utilization. After examining the numbers, you notice that utilization never exceeds 87% for any peak time on any server. You then decide to set the CPU threshold at 90% and continue to have your monitor check every five minutes, but you lower the alert threshold to sustained 90% for more than two hours. This means that if a system's CPU utilization exceeds 90% for more than two hours, you will receive a notification. That threshold for this environment is reasonable and manageable. Your level of tolerance for adding a new system to the farm, after a few months of observation, is CPU above 95% for more than two hours.

This is the process of determining how busy is busy and your tolerance levels for each service. It seems arbitrary, but it isn't, because you are observing the data on a continuous basis and making adjustments and decisions based on your observations. 90% utilization for two hours isn't excessive, but you don't want to exceed that level because then users start suffering long wait times when pulling data from your system.

Fifth: Performance alerting

I've discussed alerting but have yet to give you a solution for creating and handling events (alerts). You could create something as simple as a Bash script to check sar data for utilization numbers, but you can also deploy a commercial solution or even something in between. I won't recommend an alerting solution to you, but there are a handful of open source monitoring and alerting applications available. Most are agent-based, so add installing and maintaining another service to your list of administration duties.

As stated in the previous section, you'll have to adjust your thresholds and tolerances to avoid driving yourself crazy with alerts, especially if those alerts come across as texts to your phone(s). You only want to be notified if something is down or in trouble and requires your attention for a resolution.

The capacity planning dilemma

The dilemma of capacity planning and performance monitoring these days is that rather than purchasing several racks of servers, you're probably either leasing your server hardware, or you're using some sort of cloud solution where capacity and performance are dynamically handled by business rules. Three-year hardware leases dictate that you go through a hardware refresh every three years, whether you need to or not. The type of hardware policy you have at your company certainly alters the course of how you plan for capacity changes.

If you lease, you'll still need to perform performance monitoring and think about capacity, because if you've undersized or under-bought hardware, you'll certainly need to know. If you purchase, then you should look at performance and capacity planning on a five-year rolling basis. I say five years because managers and business owners don't want to replace hardware every three years if they're purchasing it. It's likely that they're using the hardware purchases as an amortized asset.

The trick with purchased assets is that you don't want to waste capacity by buying too much too soon. There are many stories circulating about people who buy top-of-the-line systems only to refresh them in five years without ever tapping the capacity of those systems, and after five years they're too old to bother updating and upgrading. The bottom line for acquiring new leased or purchased hardware is to buy with growth in mind and then take advantage of that growth by budgeting for upgrades based on utilization. In other words, buy what you need, upgrade as needed, and take full advantage of your hardware assets before the next refresh cycle.

Summary

Capacity planning and performance monitoring work together to give you a complete picture of your hardware and software life cycle. It's important to take the time and make the effort to set up monitoring and alerting, and to analyze the data. Too often, busy system administrators set up elaborate solutions for monitoring and then ignore them. Find a way to strike a balance between being driven crazy by performance alerts and never seeing the one that results in extended downtime. Capacity planning also helps you save money by redeploying services from overutilized to underutilized systems.