Lab 11.2: Restore Jobs
IN719 Systems Administration

Introduction

Recall that we have always talked about backups being paired with restores. If we make backup copies of our data without a plan for how to restore it and put it back into a use, there is no point to the whole exercise.Our backup procedures we established in lab 11.1 are pretty simple and our restore procedures can be simple too. But we do need to take some important things into account.

Early in the semester we considered tasks falling into four quadrants divided by a common versus rare axis and an easy versus hard axis. Backup jobs are easy and common for us, so we automate them with shell scripts and cron jobs. in comparison, restores jobs are easy and rare, so our rubric says we can do them manually. This is probably true in some sense, but there are complicating factors. First, we envision that restore jobs are going to be performed in situations where we will be under stress. Our services are degraded or not functioning and we want to fix them as quickly as possible. This extra stress pushes them to the hard side of that access. This means that we should document the process according to our guidelines. And once we document the process, we are pretty close to automating the jobs, at least to some degree.

Also, keeping in mind the pressure we anticipate when we are required to do a restore job, we can reduce that pressure by practicing the process. Best practice in this area is to perform restore drills at regular intervals. These drill both test our ability to carry out the restore jobs and test the integrity of our backup data.

In this lab we will apply these ideas and get ready to restore data when the need to do so arises.

1 Manual restore procedures

Our backup jobs just use `rsync`. Basically, an invocation of rsync looks like

```
rsync source destination
```

So the difference between an invocation of `rsync` to backup and one to restore is swapping the source and restore parameters (possibly plus or minus some options). For all of your backup `rsync` jobs, identify and test the corresponding restore jobs.

2 Document restore procedures

Now, imagine that you need to restore some files to get a service back online. You need to run the right restore commands without forgetting anything and you need to do it quickly. You might want to organise the jobs by host, e.g. "To restore the app server do the following." You might want to organise them by service, e.g., "To restore the MariaBD service do the following".

In the process of documenting, you'll probably find collections of commands that are always run as a set. I'm not saying you must write shell scripts to combine those steps into one, but that's what I would do.

3 Practice performing restores

To ensure that you can successfully restore your data when the critical time comes, you need to practice. Basically, just pick some things at random at do restore drill. Verify that the the required data was restored intact. Make sure that both members of your team do this so that either one of you is prepared for a critical outage situation. You should do these drills at least once each week for the rest of the semester.

Have you noticed that you have a ”backup” server that you’re not using for anything? You can use it to perform restore drills without worrying about disrupting the other systems. Also, that server is basically a hot standby to use if one of your other servers fails and you need to replace it. You should make plans for how you would do this.