URBackup

Installing and Configuring URBackup

https://www.youtube.com/embed/MzDmOMVUxLs

Backups are important for so many reasons. I have had backups save my bacon more than once, and I have also learned the hard way how important backups really are. Data loss is no joke, and in this day and age there is truly no reason to have data loss.

Today, I want to cover a really great backup tool that is open source, and not too difficult to get setup. URBackup (like You Are Back Up! - get it?) is a really terrific tool boasting both a Server and Client installation that then makes your backup server (where you store your data) talk to your client machines (where you are backing up your data from) with some nice auto-discovery features built in.

Now, learn from me. Make sure your firewall has port 55414 and 55415 open so that URBackup's client and server can talk easily. You'll save yourself a lot of time and headache.

What you'll need

- A server with enough drive space to store backups.
- Docker and Docker Compose installed on that server - check out https://github.com/bmcgonag/docker_installs for some easy to use scripts for various versions of Ubuntu, Debian, and CentOS.
- At least one client machine running either Windows, Linux, or both if you like.
- About an hour of time.

Installation

First, if you haven't already done so, please make sure you have both Docker-CE and Docker-Compose installed on your intended server system. This is important as it's how we will install the server portion of URBackup.

Installing Docker-CE and Docker-Compose

Installation via a Simple Script

You can easily install Docker-CE, Docker-Compose, Portainer-CE, and NGinX Proxy manager by using this quick install script I created and maintain on Github. Just use the command:

```
wget https://gitlab.com/bmcgonag/docker_installs/-/raw/main/install_docker_nproxyman.sh
```

To download the script to your desired host.

Change the permissions to make the script executable:

```
chmod +x ./install_docker_nproxyman.sh
```

and then run the script with the command:

```
./install_docker_nproxyman.sh
```

When run, the script will prompt you to select your host operating system, then will ask you which bits of software you want to install.

Simply enter 'y' for each thing you want to install.

At some point, you may be asked for your super user (sudo) password as well.

Allow the script to complete installation.

At this point, you might want to log out and back in, as this will allow you to use the `docker` and `docker-compose` commands without the need of `sudo` in front of them.

Installing URBackup Server

Now, we'll grab the docker-compose text from the URoni/URBackup dockerhub page, and setup our space for backups.

1. Setup a space to store your backups.

```
mkdir -p urbackup/{data,storage}
```

This will setup a folder structure with a new folder called "urbackup", and two folders inside of that folder called "data" and "storage".

2. Move into the "urbackup" folder:

```
cd urbackup
```

3. Create a new docker-compose.yml file:

```
nano docker-compose.yml
```

4. Paste the following code into that file:

```
version: '2'

services:
  urbackup:
    image: uroni/urbackup-server:latest
    container_name: urbackup
    restart: unless-stopped
    environment:
      - PUID=1000 # Enter the UID of the user who should own the files here
      - PGID=1000  # Enter the GID of the user who should own the files here
      - TZ=America/Chicago # Enter your timezone
    volumes:
      - ./data:/var/urbackup
      - ./storage:/backups
      # Uncomment the next line if you want to bind-mount the www-folder
      #- /path/to/wwwfolder:/usr/share/urbackup
    network_mode: "host"
    # Activate the following two lines for BTRFS support
    cap_add:
      - SYS_ADMIN
```

Save the file contents with CTRL + O, then press Enter, and exit the nano editor with CTRL + X.

5. Now run the URBackup docker-compose with:

```
docker-compose up -d
```

Once it downloads URBackup, and says "done", give it about 30 seconds, then open a web browser, and go to the ip address of your server on port 55414.

For instance, my server ip is 192.168.7.51, so I went to

```
http://192.168.7.51:55414
```

If all went well, you should now see the URBackup Server Web UI in front of you.

Install a URBackup Client

If you are using Windows, you'll download the client application .exe file from the URBackup site (https://urbackup.com). Next install the client normally, and when done, you'll use the Graphical User Interface to decide whether you want to backup only certain folders, or the entire system, etc.

You can go back to the URBackup Server, and refresh the page to see if the Windows machine has been autodiscovered. If not, you might waith a couple of minutes and try again. If it still doesn't show up, then make sure you don't have a firewall on that's blocking ports 55414 and 55415. These are the ports that URBackup uses to communicate between server and client.

Linux Client

If you are running Linux, you'll want to grab the command line command from the URBackup site, and paste it into the terminal.

At the time of writing the command is:

```
TF=$(mktemp) && wget "https://hndl.urbackup.org/Client/2.4.11/UrBackup_Client_Linux_2.4.11.sh" -O $TF && sudo sh $TF; rm -f $TF
```

But, it's always best to go to the official source, and make sure the command is up to date.

Once pasted into the terminal, run it by pressing Enter, and enter your sudo password when prompted.

You may also have to enter 'Y' at some point, and choose how you want backups / snapshots to be done.

After everything runs, check the Web Interface, and make sure your machine shows up.

Now, you can enter a command to tell URBackup what to backup.

I chose only my Downloads folder, but give it whatever path you want.

```
sudo urbackupclientctl add-backupdir -d /home/brian/Downloads/
```

Run this as many times as you like to add multiple folders for backup.

Support My Efforts on Patreon

https://www.patreon.com/bePatron?u=234177