

UGMA

Un-Google My Android

An introduction to backing up your Android device to your own cloud, not to Google.

For better layout, you can download the [LibreOffice version](#) or the [PDF version](#) of this document.

NOTE: this is not (quite) a newbie guide. You will need at least some basic experience with the Linux command-line. However: there should be enough here that with a bit of Googling and some persistence, you should be able to get it all working. Grab your nearest Linux guru and have them on standby, just in case. ;-)

NOTE: this is a work in progress, and will probably be updated every day or two while I fix formatting etc. If you're willing to read through the list of commands to execute, you can probably make it work anyway. If you have little or no Linux troubleshooting experience, come back in a day or two.

1 Introduction

This guide will help you set up a fresh installation of LineageOS on a device and connect it to a new CentOS Linux instance hosted by Digital Ocean. From what you learn doing this, and with some additional information in this guide and elsewhere, you should be able to adapt this configuration to suit your needs, your specific device, or your preferred Linux hosting environment.

1.1 Goals

The goals of the setup are:

1. Back up your Android device (enable online backup and restore of a single Android device)
2. Synchronise your Android device (enable synchronisation of your data & files between Android devices)
3. Access your Android device's data online (enable online access to your data and files)

Not all software can easily be backed up with this system, so you may have to experiment and adapt it to suit your needs depending on what apps you have running on your device.

1.2 Requirements

- An email address (for SSL certificate registration)
- A discount code for a DO droplet (sign up here for USD10 credit: <https://m.do.co/c/463c92da0f84>)

- A device capable of running LineageOS
- A domain name (or DDNS service maybe?) (for your SSL certificate)

1.3 Assumptions

This documentation makes the following assumptions about your skills:

- Your Linux skills:
 - You can generally use a Linux terminal (eg: issue commands)
 - You can navigate a filesystem within a Linux terminal (eg: “cd /var/www”)
 - You can edit a text file within a Linux terminal (eg: “vim ssl.conf”, “emacs ssl.conf”, “nano ssl.conf”, etc)
- Your Android skills:
 - You can generally use the Android interface
 - You can find your way around the system settings

1.4 Typographic conventions

Some things to note:

Format	Purpose
<code>literal text</code>	Denotes the name of a file, text you must place into a file, or a command you should type into a Linux console (depending on context). In most cases, you could simply copy this and paste it into the console.
<i>Placeholder</i> (in prose); or <i>Placeholder</i> (in literal text)	Denotes something that you should replace; typically the placeholder text describes what should go there. Watch out when copy/pasting text with placeholders: be sure to replace them with something suitable!

1.5 Relax!

Although this is a long document, most of the server-side setup can be done by copy-paste, as long as you use the same environment used in the documentation (a CentOS 7 VM hosted at Digital Ocean).

2 What to back up

Here's a list of some of the things you might like backed up or synchronised from your Android device, and what this guide currently helps you achieve:

Category	Item	Status
Common user data	Calendars	Synchronised
	Contacts	Synchronised
	Phone call logs	Backed up to IMAP and a calendar
	SMS messages	Backed up to IMAP
	Photos/videos	Synchronised
	Music	Not yet done...
	Browser bookmarks	Not yet done...
	Browser passwords	Not yet done...
	General passwords	Not yet done...
	Other	Not yet done...
Application-specific data	Application files (APKs)	Not yet done...
	Application preferences	Not yet done...
	Application data	Not yet done...
System data	System accounts	Not yet done...
	System preferences	Not yet done...

3 Recording important information

Here's a table to print and fill in while going through this process. Mostly, you'll want to store passwords here. Be sure to generate some secure ones! If you want truly random ones, consider using [GRC's password generator](#).

Server information	
IP address	
Hostname	
Root password	(n/a – use SSH-based password-less login)
Database management information	
Database management password	
Database configuration	
Nextcloud database user name	nextcloud
Nextcloud database user password	
Nextcloud database	nextcloud
Nextcloud administrator user	
Nextcloud admin user name	admin
Nextcloud admin user password	
Nextcloud user (ie: your normal Nextcloud account)	
Nextcloud normal user name	
Nextcloud normal user password	
Mail account for SMS backups (ie: IMAP account)	
SMS user name	
SMS password	

4 Setting up the server

4.1 Installing the operating system (CentOS 7.x)

Any hosting environment would suffice, be it physical or virtual. If you want to set up a virtual environment, try Digital Ocean they have VMs for as little as USD5 per month.

Here's a discount code to get you USD10 credit with Digital Ocean (enough for two months of free hosting): <https://m.do.co/c/463c92da0f84>.

Installing CentOS is out of the scope of this documentation, but with Digital Ocean it's just a few clicks away. Even better, you only pay for your hosting by the hour, which makes for great experimentation for just a few cents! As long as you destroy any experimental VMs when you're done with them, they'll cost you next to nothing.

The rest of the commands in this section should be done at a root prompt on the virtual machine.

4.1.1 Why CentOS?

This is almost entirely because I'm most familiar with CentOS; you could just as easily use Debian, Ubuntu, Fedora, Suse, or just about any other modern Linux server. You'll have to substitute “yum” for “dnf”, “apt-get”, or other software-management tools specific to your chosen distribution. Also, some steps may not be required, and other additional steps may be required, depending on the default set of software installed. Also, some commands and file content may need to differ from what I've presented here.

4.1.2 Why Digital Ocean?

They are convenient, easy-to-use, provide a management API, and offer a USD5/month plan that is perfectly adequate for starting your own cloud server. There are many other providers (eg: Linode); feel free to shop around and choose something to your liking.

4.2 Preparing the host

We'll be installing a few things that make the process a little easier, and setting up basic server settings.

- Update your OS:
 - `yum -y update`
- Set regional settings (substitute *Region* and *Location* for your region and location, and *en_GB.utf8* for your preferred language):
 - `ln -f -s /usr/share/zoneinfo/Region/Location /etc/localtime`
 - `localectl set-locale LANG=en_GB.utf8`

- Install useful tools
 - `yum -y install screen wget unzip`
- Install a text editor (choose one of these, based on your preferred text editor):
 - `yum -y install vim`
 - `yum -y install pico`
 - `yum -y install emacs`
- Install 3rd-party repos (for CertBot and PHP 7):
 - `rpm -Uvh https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm`
 - `rpm -Uvh https://mirror.webtatic.com/yum/el7/webtatic-release.rpm`
- Disable SELinux
 - Edit `/etc/selinux/config` and update the line that starts with `SELINUX=` to look like this:
 - `SELINUX=permissive`
- Reboot (ensure latest kernel is running, regional settings are set correctly, and SELinux is disabled):
 - `reboot`

Further commands can be run inside a screen session; this way, if you are disconnected from the server then any command you were running at the time won't be aborted.

4.3 Installing the web server

We'll be installing Apache 2, and PHP 7. Apache will be configured with SSL support.

- Install Apache & PHP7:
 - `yum -y install httpd mod_ssl mod_php71w php71w-cli php71w-common php71w-gd php71w-mbstring php71w-mcrypt php71w-mysqlnd php71w-xml php71w-posix`
- Turn off the web-server welcome message:
 - `rm -f /etc/httpd/conf.d/welcome.conf`
 - `echo "# keep this file here to prevent updates from re-enabling the server welcome message" >/etc/httpd/conf.d/welcome.conf`

- Enable and start Apache:
 - `systemctl enable httpd`
 - `systemctl start httpd`

4.4 Setting up an SSL certificate

Thanks to the Let's Encrypt project, it's now trivial to generate your own free SSL certificates. We'll install CertBot to do the grunt work, and enable SSL in Apache.

- Install CertBot:
 - `yum -y install certbot`
- Generate certificate (replace `cloud.yourdomain.com` with your VM's hostname):
 - `certbot certonly --register-unsafely-without-email --webroot --webroot-path /var/www/html --agree-tos -d cloud.yourdomain.com`
 - Note that this command generates a certificate without using an email address. You really should use an email address for account recovery and other purposes: you can run the command `certbot certonly` for a fully interactive experience.
- Replace or add these lines (look for lines that start with the same keyword) in `/etc/httpd/conf.d/ssl.conf` (replace `cloud.yourdomain.com` with your hostname):
 - `SSLCertificateFile /etc/letsencrypt/live/cloud.yourdomain.com/cert.pem`
 - `SSLCertificateKeyFile /etc/letsencrypt/live/cloud.yourdomain.com/privkey.pem`
 - `SSLCertificateChainFile /etc/letsencrypt/live/cloud.yourdomain.com/chain.pem`
- Restart webserver:
 - `systemctl restart httpd`

4.5 Installing IMAP

IMAP is a protocol that can be used to access your email on a server. We'll be setting it up so that we can back up our SMS messages and phone call logs.

- Install dovecot for SMS backup:
 - `yum -y install dovecot`

- Configure certificate for secure IMAP access:
 - Edit `/etc/dovecot/conf.d/10-ssl.conf` and modify these lines (look for the keywords at the start of the lines; replace `cloud.yourdomain.com` with your hostname):
 - `ssl_cert =`
`</etc/letsencrypt/live/cloud.yourdomain.com/fullchain.pem`
 - `ssl_key =`
`</etc/letsencrypt/live/cloud.yourdomain.com/privkey.pem`
- Configure dovecot's mailstore detection:
 - Edit `/etc/dovecot/conf.d/10-mail.conf` and change the line starting with `mail_location =` to (add the line if it's not already there, or uncomment it if it starts with "#"):
 - `mail_location = mbox:/home/%u/mail:INBOX=/var/spool/mail/%u`
- Enable and start dovecot:
 - `systemctl enable dovecot`
 - `systemctl start dovecot`
- Add a Linux user for SMS:
 - `useradd sms`
 - `passwd sms`
 - (Enter the password you set for the SMS mailbox)

Although we haven't shown how to do so here, it's possible to connect a mail client (such as Thunderbird) to the IMAP service on your mail server in order to access your SMS messages or organise them into folders. Note that if you organise them into folders, they won't be restored to your phone if you restore to a new or factory-reset device.

4.6 Installing the database software

Although Nextcloud supports the use of a convenient database called Sqlite, you'll get scalability and better performance by using MySQL or a fork of it known as MariaDB. We'll be installing MariaDB, securing it, and preparing it for Nextcloud.

- Install database, enable it, start it, and secure it:
 - `yum -y install mariadb-server`
 - `systemctl enable mariadb`
 - `systemctl start mariadb`

- `mysql_secure_installation`
 - Follow all recommended defaults; do not miss this part and do not skip any steps.
- Create database instance and database user (replace `nextcloud-database-password` with a new database password):
 - Run `mysql -u root -p` then enter the following commands into it:
 - `CREATE DATABASE nextcloud;`
 - `CREATE USER nextcloud@localhost IDENTIFIED BY 'nextcloud-database-password';`
 - `GRANT ALL PRIVILEGES ON nextcloud.* to nextcloud@localhost;`
 - `FLUSH PRIVILEGES;`
 - `EXIT`
- Enable Binary Logging in MariaDB:
 - Edit the database config file `/etc/my.cnf` and add the following three lines in the `[mysqld]` section:
 - `log-basename=master`
 - `log-bin`
 - `binlog-format=mixed`
 - Restart MariaDB:
 - `systemctl restart mariadb`

4.7 Installing Nextcloud

Now is the interesting part! Here's where we finally install Nextcloud. You should run these commands either from root's home folder (`/root`) or from the web-server's home folder (`/var/www`)

- Download, verify, and authenticate nextcloud files:
 - `wget`

```
https://download.nextcloud.com/server/releases/nextcloud-13.0.0.zip
```
 - See https://docs.nextcloud.com/server/13/admin_manual/installation/source_installation.html for verification information, if you'd like to verify the downloaded files (recommended for production!)

- Unzip nextcloud files and move to web root:
 - `unzip -q nextcloud-13.0.0.zip`
 - `mv nextcloud/* nextcloud/.??* /var/www/html`
 - The peculiar pattern “.??*” ensures that normally-hidden web-server configuration files are also moved, but without causing issues related to the fact that “.” and “..” are special folders in Linux
- Set bare-minimum permissions to permit functionality (WARNING: must secure this properly later):
 - `chown apache.apache -R /var/www/html/`
 - `chmod +x /var/www/html/occ`
 - `mkdir /var/www/data`
 - `chown apache.apache /var/www/data`
 - `chmod 750 /var/www/data`
- Set strong directory permissions:
 - TODO: fill this in...
- Go to web browser and complete installation, with the following specific settings:
 - Set the data folder to `/var/www/data`
- Log in as admin and perform other useful setup:
 - Check Settings → Basic settings to see if there are any issues that must be resolved
 - Enable useful App Bundles:
 - Groupware Bundle
 - Add a user for yourself
- Log out, then log in as your normal user and change settings to taste; eg:
 - Personal info:
 - Set the language (eg: en_GB to get Monday as the first day of the week)
 - Optional: add a calendar to record phone calls (SMS Backup+ can log all calls to a calendar for you)

4.8 Post-install tasks

- Import any calendars and/or contacts that were exported from other sources

5 Setting up your device

5.1 Preparing your device

Note that this is optional; if you are happy to use whatever Android installation comes with your device, go right ahead and skip to the next sub-section. Note that some devices have been configured in such a way that you actually must add a Google account before using the device. This seems to be something that the manufacturer can control, since not all devices have this requirement.

Here's the recommended process for preparing your device (note that the steps are device-specific, and are beyond the scope of this documentation; also, you may choose to skip some steps or do something entirely different in order to achieve the same thing):

1. Download and install ADB and Fastboot for your desktop computer (Google can help you find the right version for Windows; if you use a Linux desktop, your distro's software installation tool can probably help)
2. Download [TWRP](#) and [LineageOS](#) for your device
3. Unlock your device's bootloader
4. If you want to back up your device's stock recovery before replacing it with TWRP, use `fastboot boot` to run TWRP before installing it and back up the recovery partition
5. Install TWRP
6. Back up the factory ROM
7. Copy the recovery and ROM backups from your device to somewhere else
8. Install LineageOS

During the first boot (either after installing LineageOS or after a factory reset), consider enabling Privacy guard; this blocks any attempts by apps from accessing your private data. It can be a bit irritating the first time each app asks for access to something, but you typically choose "always", "never", or "ask me each time"; by choosing "always" or "never", you're only asked the once for each permission for each app.

5.2 Installing packages

- In Settings → Security:
 - Enable Other Sources

- Install F-Droid
 - Install from <https://f-droid.org/>
- Update F-Droid
 - Open F-Droid
 - Wait for the repositories to be updated
 - If there's an update to F-Droid, install it
- Install various useful packages from F-Droid:
 - Nextcloud
 - DAVdroid
 - SMS Backup+

5.3 Connecting your device to Nextcloud

This sets up file synchronisation, so that you can share files between your device(s) and your server.

- Initial setup:
 - Open the Nextcloud app
 - Skip through the info to the setup page
 - Enter the server address (include <https://> at the front)
 - Grant access to the account to your device
 - Log in with username and passwords
 - Allow Nextcloud access to photos, media, and files
- Other useful settings:
 - Enable auto upload of photos
 - TODO: how??

5.4 Setting up calendar and contact synchronisation

This connects Android's built-in calendar and contact apps to your Nextcloud server.

- Open the Nextcloud app
- In Settings, select Sync calendar & contacts

- You'll be sent to the "Add account" dialog of DAVdroid
- Enter the password for your Nextcloud account (everything else should be pre-filled for you) and click Login
- Name the account as you prefer, and select "Groups are per-contact categories" as the Contact group method
- Go to DAVdroid from the DAVdroid permissions notification
- Grant the requested permissions
- Go to DAVdroid
- Review Battery Optimisation (you probably want to turn Battery Optimisation off for DAVdroid), Background running (Android O and later) and OpenTasks notices
- Open the account you named above
- Tick all CardDAV and CalDAV feeds
- Change account settings:
 - Enable automated sync (select a sync interval) for calendar and contacts
- Synchronise the account (will fetch calendar and contacts from the server)

5.5 Setting up SMS and call-log backup

SMS Backup+ can back up your SMS messages and phone call logs to IMAP folders. SMS Backup+ can also save phone call logs to a calendar, making it easy to see your call history in Nextcloud or on your device.

- Open SMS Backup+
- Grant permissions (skip calendar if you don't want to add calls to a calendar)
- Set up Advanced settings:
 - Set up IMAP server settings:
 - Change Authentication to Plain text
 - Set the server address to your Nextcloud server (keep the :993 port number)
 - Set the username and password to the SMS mail username and password
 - Set up Backup settings:
 - Tick Backup SMS, Backup MMS, and Backup Call log
 - Update Call log settings:

- Set Call log label to the name of the mail folder in IMAP where you want to store call logs (the default of “Call log” will probably suffice)
- Set Call types to Everything (or change to your taste; the default should be Everything)
- Enable Calendar sync (if you want to record calls in a calendar)
- Select a calendar for Calendar synchronisation
- Set backup after call
- Set Maximum items per backup to All (this should be the default)
- Set Mark as read (emails) to Use message status
- Set Gmail label to the name of the mail folder in IMAP where you want to store SMS messages (the default of “SMS” will probably suffice)
- Set Contacts to back up to Everybody
- Set Email address style to suit your personal taste (“Name (+00 123456)” is recommended)
- Set up Restore settings:
 - The defaults are all ticked (except “Only restore starred items”); change to suit your taste
 - Change the number of latest messages to restore to suit your taste
- Enable Notifications and Confirm actions
- Make a call or send an SMS, so there’s something to back up
- Perform a Backup, just to check that it all connects and works (if this is a clean device, it won’t matter; if it has messages on it, you may care to choose “Backup” when it asks whether to back up or skip messages currently stored on the device)
 - Allow access to things that it needs access to
 - Send an SMS and make a call, then run backup; you may be asked to grant permissions to the app to access the call log and/or SMS messages
- If automated backups are failing but manual backups work, and you’re on Android O or later, check that Battery optimisation is off for SMS Backup+ (Settings → Apps & notifications → Special app access → Battery optimisation, select “All apps”, check the list for SMS Backup+, select it, and select Don’t optimise)

6 Troubleshooting

Apache log files:

- `/var/log/httpd/ssl_error_log`

NextCloud log files:

- `/var/www/data/nextcloud.log`

Test connection to imaps:

- `openssl s_client -connect cloud.yourdomain.com:imaps`
 - `?LOGIN username password`
-

7 Maintenance

7.1 Backing up your cloud-hosted data

7.2 Other?

8 Recovery and other tasks

8.1 Recover from server failure

8.2 Recover from phone loss or failure

8.3 Migrate cloud-hosted data to a new server

9 Security considerations

It might be better to configure SELinux properly, rather than disabling it. Here's a possible starting point for this: <https://www.marksei.com/install-nextcloud-12-centos-7/>

10 Additional notes

10.1 Hosting notes

In general, you should be able to set up online backup/sync/access of your device data to almost any type of hosting arrangement in almost any location.

10.1.1 Hosting locations

Here are some possible hosting locations:

- So-called “live” hosting, available from anywhere in the internet (ie: “online”)
- Internal hosting, no external access (sync and backup will only work while on the same network; also, see below about SSL certs)
- Internal hosting with VPN access (sync and backup will only work while connected to your home or office via VPN, but should work from anywhere; also, see below about SSL certs)
- Internal hosting with DMZ or port forwarding from outside (more or less equivalent to “live” online hosting; also, see below about SSL certs)

SSL note: Note that internal-only or VPN-only access might make it difficult to issue and maintain SSL certificates via Let’s Encrypt since Let’s Encrypt needs to reach your server in order to verify the hostname. You may still be able to buy and use a certificate using a traditional SSL certificate provider, though. Let’s Encrypt should work OK with DMZ or port-forwarding configurations.

Internal hosting note: If you host Nextcloud internally with access externally (either via DMZ or port forwarding), you may have issues connecting internally using the external hostname. If this is the case, either find a firewall/router that supports [hair-pin routing](#), or set up the hostname to resolve internally to the correct internal IP address.

10.1.2 Hosting environments

Nextcloud and other tools used in the server side of this setup should work on just about any conceivable modern Linux distribution, and could probably be made to work adequately on Windows.

10.1.3 Server configurations

This setup is hardware-agnostic, and so could work on any of these types of system and more:

- Physical server
- Virtual server
- Docker container
- VirtualBox virtual machine
- Digital Ocean, Linode, etc. virtual machine
- Raspberry Pi

10.1.4 Multi-device and multi-user arrangements

You could add users to Nextcloud and/or IMAP to support multiple devices and/or multiple users. Here are a few possibilities:

- For multiple device support, back up photos from each device into different folders in Nextcloud and back up SMS and call logs to different IMAP folders for each device (prevents photo filename collisions between devices, which would overwrite each other's photos)
- Add Nextcloud users and IMAP users to support a single server among family members, a group of friends, or with work colleagues
- Set up an online service set up to sell or rent UGMA services to people (you'll probably want to automate a bunch of stuff to make this work well, especially if you want this to scale up)

10.2 References

Server setup stuff:

- Start here: <https://community.time4vps.eu/discussion/110/nextcloud-nextcloud-12-0-installation-with-mariadb-and-php-7-1>
- This has good security advice, too: <https://www.marksei.com/install-nextcloud-12-centos-7/>
- Another one to check out: <https://cloudpro.zone/index.php/2017/11/16/installing-nextcloud-12-centos-7-4/>
-

Server setup caveats:

- This setup configures the whole webspace to be owned by apache; it should be owned by root, readable by apache, and specific folders writable by apache. Or something like that. See these two pages for possible alternatives to improving this setup:
 - <https://www.marksei.com/install-nextcloud-12-centos-7/>
 - <https://community.time4vps.eu/discussion/110/nextcloud-nextcloud-12-0-installation-with-mariadb-and-php-7-1>

11 Future improvements

- Show how to set up redirection from http:// to https://
- Install git (for saving passwords via the “pass”-compatible PwdStore Android app)
- Fill in the sections on troubleshooting, maintenance, and recovery
-

