Gensyn Node Setup Guide

This document contains a full step-by-step guide to setting up a Gensyn node on a fresh Ubuntu server. Each step includes the exact command and a short explanation of what it does.

---

1. Update Package Lists and Install `sudo` (if not available)

```
sudo apt update && sudo apt install -y sudo
```

Explanation: Updates the system's package list and installs the `sudo` command if missing.

---

2. Install Essential Tools

```
sudo apt update && sudo apt install -y python3 python3-venv python3-pip curl
wget screen git lsof nano unzip iproute2 build-essential gcc g++
```

Explanation: Installs all necessary packages including Python, networking tools, compilers, editors, and build tools.

---

3. Install CUDA Dependencies (Optional - GPU acceleration)

```
[ -f cuda.sh ] && rm cuda.sh; curl -o cuda.sh https://
raw.githubusercontent.com/zunxbt/gensyn-testnet/main/cuda.sh && chmod +x
cuda.sh && . ./cuda.sh
```

Explanation: Downloads and runs a CUDA setup script to enable GPU compute (if needed).

---

4. Reinstall Basic Dependencies (Redundancy Check)

```
sudo apt update && sudo apt install -y python3 python3-venv python3-pip curl
wget screen git lsof
```

Explanation: Ensures the critical packages are installed properly.

---

5. Install Node.js (v20)

```
curl -fsSL https://deb.nodesource.com/setup_20.x | sudo -E bash -
sudo apt update && sudo apt install -y nodejs
```

Explanation: Adds NodeSource repository for Node.js v20 and installs Node.js.

---

6. Install Yarn Package Manager

```
curl -sS https://dl.yarnpkg.com/debian/pubkey.gpg | sudo apt-key add -
echo "deb https://dl.yarnpkg.com/debian/ stable main" | sudo tee /etc/apt/
sources.list.d/yarn.list
sudo apt update && sudo apt install -y yarn
```

Explanation: Adds Yarn's package repository and installs it.

---

7. Start a New Screen Session

```
screen -S gensyn
```

Explanation: Opens a new screen session named `gensyn` for background task persistence.

---

8. Clone the Gensyn Repository

```
git clone https://github.com/gensyn-ai/rl-swarm.git && cd rl-swarm
```

Explanation: Clones the official Gensyn `rl-swarm` repository and enters the directory.

---

9. Create and Activate Python Virtual Environment

```
python3 -m venv .venv
source .venv/bin/activate
```

Explanation: Creates a virtual environment in `.venv` folder and activates it.

---

10. Install Frontend Dependencies

```
cd modal-login
yarn install
```

```
yarn upgrade
yarn add next@latest
yarn add viem@latest
```

Explanation: Installs and upgrades frontend JavaScript dependencies using Yarn.

---

11. Reset and Update Git Repo (Optional Recovery)

```
cd ..
git reset --hard
git pull origin main
git checkout tags/v0.5.1
```

Explanation: Resets local changes, pulls the latest code from GitHub, and checks out version `v0.5.1`.

---

12. (Optional) Transfer Your Swarm File

If you have a previous account, upload your swarm file into the VPS using `scp` or another method.

---

13. Run Gensyn Node

```
cd $HOME/rl-swarm/
python3 -m venv .venv && . .venv/bin/activate && ./run_rl_swarm.sh
```

Explanation: Activates virtual environment and starts the Gensyn node.

✅Done!

Your Gensyn node should now be running. To detach from the screen session:

```
Ctrl + A, then D
```

To reattach later:

```
screen -r gensyn
```

GitHub Repo Suggestion

You can create a GitHub repository with this README content. Example repo structure:

```
Gensyn-Node-Setup/
├── README.md   <-- This guide
└── run_instructions.sh  <-- Optional shell script version of all commands
```

Let me know if you'd like the `.sh` version too.