rsync — инкрементный бэкап на удаленный сервер

SDATA (источник)
User: user1
Password: pass1
Директория для переноса: /home/k138/

SERVER2 (приемник)
User: user2
Password: pass2
Директория для приема информации: /home/user2/_backup/SDATA

Логинимся через SSH на SDATA.

$ ssh-keygen -t rsa

путь по умолчанию ./.ssh/id_rsa затем два раза Enter.

Копируем ключ на сервер:

$ ssh-copy-id user2@server2

Если файл *authorized_keys* не существует, то он автоматически будет создан, проверяем права:

-rw------- 1 user2 user2 5164 Май 30 2020 authorized_keys

Если необходимо меняем на *600*.

Далее используем команду *rsync* с первого сервера (SDATA) для проверки с выводом информации о прогрессе:

$ rsync -avz --progress /home/k138/ user2@server2:/home/user2/SDATA

рекомендуется использовать *nice* что бы процесс копирования шел в фоновом режиме:

$ nice -n +17 rsync -avz /home/k138/ user2@server2:/home/user2/SDATA

Пишем скрипт для крона *rsync_backup_sdata_to_server2.sh*, в котором лучше прописать полные пути до *nice* и *rsync*. В моем случае пути такие:

```
#!/bin/bash
/usr/bin/nice -n +17 /usr/bin/rsync -avz /home/k138/ user2@server2:/home/user2/SDATA
```

Создаем задание в cron:

```
$ crontab -e
```

```
30 5 * * * /bin/sh /home/user1/bin/rsync_backup_sdata_to_server2.sh
```

Эта команда означает, что бэкап будет выполняться каждый день в 5-30 утра.