

Module 1: Server Management Basics

This module provides an overview of essential server management tasks in CWP, including managing server settings and services, basic system administration commands, monitoring resources, and scheduling tasks using cron jobs.

1. Managing Server Settings and Services

CWP allows you to configure and manage key server settings and essential services through its interface. Proper management of these settings is crucial for server performance and security.

Accessing Server Settings:

- In the CWP Admin Panel, go to **Server Settings** to adjust various configurations:
- They include
- root cron jobs

Crontab for user root..

Don't forget to always use a full path, eg. /usr/local/bin/php

List of active cron jobs

Cron Number	Cron Job	Delete
1	00 03 * * * /usr/local/cwp/php71/bin/php -d max_execution_time=18000 -q /usr/local/cwpsrv/htdocs/resources/admin/include/cron_autoss_l_all_domains.php	Delete
2	0 0 * * * /usr/local/cwp/php71/bin/php -d max_execution_time=18000 -q /usr/local/cwpsrv/htdocs/resources/admin/include/alertandautorenewssl.php	Delete
3	26 1 * * * "/root/.acme.sh"/acme.sh --cron --home "/root/.acme.sh" > /dev/null	Delete

Add Common Cron Jobs

Settings:

Command:

Save changes

```
# Example of job definition:
# .----- minute (0 - 59)
# | .----- hour (0 - 23)
# | | .----- day of month (1 - 31)
# | | | .----- month (1 - 12) OR jan,feb,ma
# | | | | .---- day of week (0 - 6) (Sunday=
# | | | | |
# * * * * * user-name command to be executed
```

Add Custom Cron Jobs

- users cron jobs

crons You are here:

Scheduled Cron Jobs [New cron](#)


Filter by account: Search:

#	Account	Command	When runs	Actions
1	lintsaw	kill -9 \$(ps faux grep python grep -v grep awk '{print \$2}')	Once per minute	
2	root	/usr/local/cwp/php71/bin/php -d max_execution_time=18000 -q /usr/local/cwpsrv/htdocs/resources/admin/include/cron_autoss_l_all_domains.php	(00) (03) (*) (*) (*)	
3	root	/usr/local/cwp/php71/bin/php -d max_execution_time=18000 -q /usr/local/cwpsrv/htdocs/resources/admin/include/alertandautorenewssl.php	Daily (at midnight)	
4	root	"/root/.acme.sh"/acme.sh --cron --home "/root/.acme.sh" > /dev/null	(26) (1) (*) (*) (*)	

Showing 1 to 4 of 4 entries Previous Next

- changing root password

CWP7 admin Dashboard Terminal FileManager Support root Logout

Load: 0.00 0.00 0.00 rootpwd You are here: 

Navigation

Search..

- Dashboard
- CWP Settings
- Server Settings
 - Crontab for root
 - Crontab for users
 - Change Root Password**
 - SSH Key Generator
 - Change Date & Time
 - Change Hostname
 - Disk Quota
 - Yum Manager
 - Reboot Server

Change Root Password

Please don't use special characters!!!

New Password: (random generator)

Confirm New Password:

[Change Root Password](#)

- Changing hostname

Search..

Dashboard CWP Settings Server Settings

- Crontab for root
- Crontab for users
- Change Root Password
- SSH Key Generator
- Change Date & Time
- Change Hostname**
- Disk Quota
- Yum Manager
- Reboot Server

WebServer Settings PHP Settings Service SSH

Change Hostname

Hostname should be only as a subdomain, like: srv1.mydomain.com, please don't use cloudflare protection with hostname as this will cause you issues. If you are using a VPS with OpenVZ/Virtuozzo/Lxc containers then you should also update hostname within the VPS panel.

Hostname change will also generate a new Hostname autoSSL Certificate.
 Certificate Path: [/etc/pki/tls/certs/hostname.bundle](#)
 Key Path: [/etc/pki/tls/private/hostname.key](#)
 Pure-FTPd PEM: [/etc/pki/tls/private/hostname.pem](#)

Your Hostname is: **srv2.lintsawa.com** and it resolves to IP: **151.80.93.107** [[Check Black List](#)] [[Check CWP SSL](#)] [[Check WebServers SSL](#)]
 rDNS/PTR = **ip107.ip-151-80-93.eu** SUCCESS [[Check SenderBase](#)]

rDNS/PTR check for IP **151.80.93.107** = ip107.ip-151-80-93.eu

CWP Admin Panel Link (by hostname)
 CWP Admin Panel Link: <http://srv2.lintsawa.com:2030>
 CWP Admin Panel Link: <http://srv2.lintsawa.com:2086>
 CWP Admin Panel SSL Link: <https://srv2.lintsawa.com:2031>
 CWP Admin Panel SSL Link: <https://srv2.lintsawa.com:2087>

CWP User Panel Link (by hostname)
 CWP User Panel Link: <http://srv2.lintsawa.com:2082>
 CWP User Panel SSL Link: <https://srv2.lintsawa.com:2083>

Change Hostname c + x

New Hostname:

Enter New Hostname, it will also generate a new SSL

- Generating ssh keys

Load: 0.26 0.11 0.04

Navigation

Search..

Dashboard

CWP Settings

Server Settings

- Crontab for root
- Crontab for users
- Change Root Password
- SSH Key Generator**
- Change Date & Time
- Change Hostname
- Disk Quota
- Yum Manager
- Reboot Server

ssh_KeyGenerator

You are here: 🏠

SSH Key Access Generator

Private Key is used for accessing the server from remote computer.

Public Key is used to allow remote connection on the local server
Place the public key on the server that you want to access, in file `/root/.ssh/authorized_keys`.

If you want to access this server with the ssh key than copy Public key to `/root/.ssh/authorized_keys` and copy Private key to your computer which you will use for accessing this server, Private Key location `/root/.ssh/id_rsa`.

If you want to access multiple servers with one key than copy the same Public Key to all servers.

Your Authorized Keys which have access to this server from: `/root/.ssh/authorized_keys`

```
ssh-ed25519 AJ...JT8RoYIjYNq...ll.com
```

Generate New Keys
Generate New SSH Private and Public Key, It will overwrite old files if they exist.

[Generate New Key's](#) [Add Public Key to Authorized](#)

- Changing date and time

Load: 0.12 0.13 0.05

Navigation

Search..

Dashboard

CWP Settings

Server Settings

- Crontab for root
- Crontab for users
- Change Root Password
- SSH Key Generator
- Change Date & Time**
- Change Hostname
- Disk Quota
- Yum Manager
- Reboot Server

date_time

You are

Change Server Date & Time

This server date & time: Tue Nov 12 00:59:18 EST 2024
Your browser date & time: 12/11/2024 09:00:05 [Apply this date & time bellow](#)

Date *:	<input type="text" value="12/11/2024"/> (DD/MM/YYYY)
Time *:	<input type="text" value="05"/> : <input type="text" value="59"/> (24h Format)
Timezone:	<input type="text" value="America/New_York"/> (Don't forget to change <code>/usr/local/php/php.ini</code> and <code>/usr/local/cwp/php71/php.ini</code> timezone (" <code>date.timezone</code> " field))

* This option do not work on VPS systems like openVZ, for openVZ Date/Time need to be set on the main node but you can change the timezone.

[Change date & time](#)

- Disk Quota Settings

Disk Quota Settings

If you are using a VPS server then you should contact your hosting provider for the quota setup.
[Instruction on how to setup quota](#)

Your disk status from the file /etc/fstab

```
#
# /etc/fstab
# Created by anaconda on Wed Mar 31 03:00:24 2021
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
#UUID=ce96c176-d129-4a36-b7c6-cdc545d84746 /          ext4    defaults    1 1
LABEL=root / ext4 defaults 0 0
/dev/vda2 swap swap defaults 0 0
```

Your Partition scheme, output of the command df -h

Filesystem	Size	Used	Avail	Use%	Mounted on
devtmpfs	848M	0	848M	0%	/dev
tmpfs	866M	0	866M	0%	/dev/shm
tmpfs	866M	76M	791M	9%	/run
tmpfs	866M	0	866M	0%	/sys/fs/cgroup
/dev/vda1	58G	4.9G	50G	9%	/

- Server reboot.

Reboot Server Now

Reboot Server

[Reboot Server Now](#)

Reboot Server in number of Minutes (eg. 60 minutes = 1 hour)

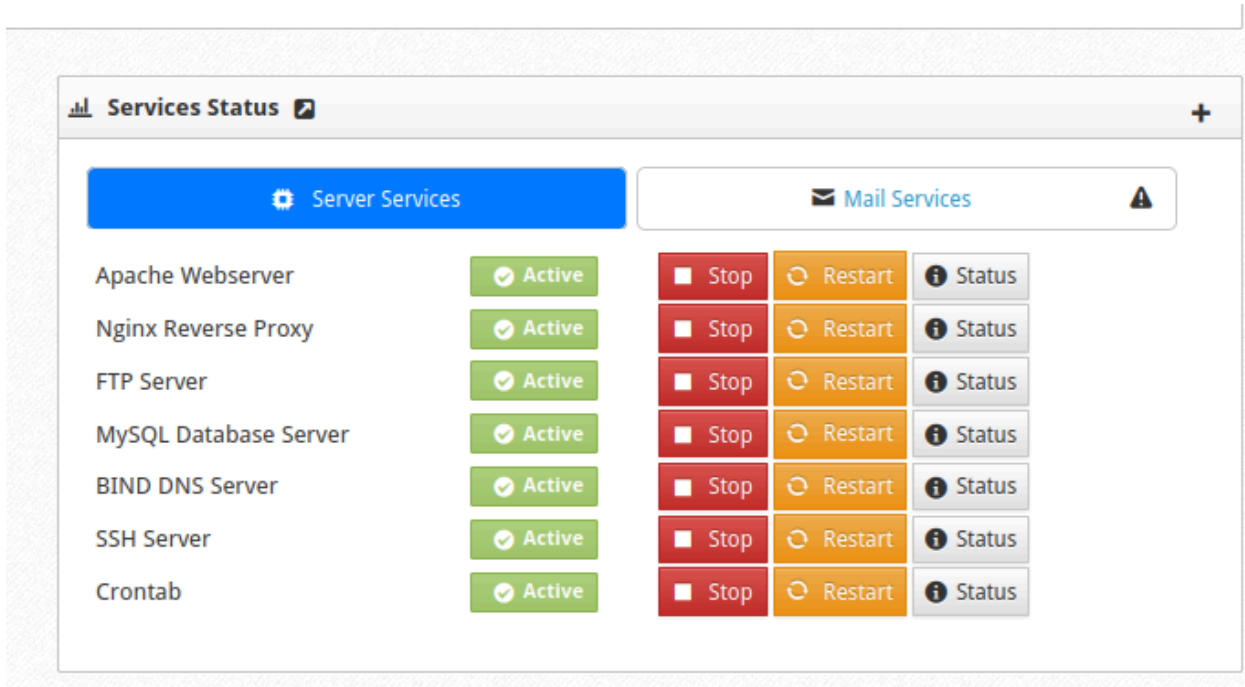
Reboot Server in: (number of minutes)

[Reboot in](#)

Managing Services:

- **Service Manager:** Navigate to **Dashboard > Services Status** in CWP to view, start, stop, or restart services.
- **Common Services to Manage:**
 - **Apache/Nginx/HTTPD:** Web server processes.
 - **MySQL/MariaDB:** Database services.

- **FTP**: File Transfer Protocol service for managing files.
- **Dovecot/Postfix**: Email services.
- **CSF/LFD**: Security services like the firewall.
- **Bind/DNS Server** : DNS services
- **SSH Server** : SSH services



- Each service can be restarted from the CWP panel.

2. Basic Commands for System Administration

Using basic command-line tools for server management allows you to troubleshoot issues, configure settings, and monitor the system efficiently.

Common Commands for Server Administration:

- **Navigating Directories:**
 - **cd** — Change directories (e.g., **cd /var/www/html**).
 - **ls** — List files and directories.
- **File Management:**

- **cp** — Copy files or directories (e.g., **cp file.txt /backup**).
- **mv** — Move or rename files (e.g., **mv file.txt newname.txt**).
- **rm** — Remove files or directories (use with caution).
- **System Information:**
 - **uname -a** — Show system information.

```
[root@srv ~]#
[root@srv ~]#
[root@srv ~]# uname -a
Linux srv.lintsawa.com 4.18.0-553.22.1.el8_10.x86_64 #1 SMP Tue Sep 24 05:16:59 EDT 2024 x86_64 x86_64 x86_64 GNU/Linux
[root@srv ~]#
```

- **df -h** — Check disk space usage.

```
[root@srv ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        356M   0  356M   0% /dev
tmpfs           374M  512K  373M   1% /dev/shm
tmpfs           374M   34M  341M   9% /run
tmpfs           374M   0  374M   0% /sys/fs/cgroup
/dev/vda1       29G   7.3G   20G  27% /
tmpfs           75M   0    75M   0% /run/user/0
[root@srv ~]#
```

- **free -m** — Display memory usage.

```
[root@srv ~]# free -m
              total        used         free       shared  buff/cache   available
Mem:           746          193           61           20          491          415
Swap:          998           252           746
[root@srv ~]#
```

- **uptime** — Show how long the server has been running.

```
[root@srv ~]# uptime
01:35:12 up 18 days, 19:30,  1 user,  load average: 0.05, 0.10, 0.08
[root@srv ~]#
```

- **Process Management:**

- **top** or **htop** — Display running processes and resource usage.

```

CPU[|||||] 3.9% Tasks: 56, 41 thr, 55 kthr; 1 running
Mem[|||||] |200M/747M| Load average: 0.14 0.12 0.09
Swp[|||||] 248M/999M Uptime: 18 days, 19:32:16

Main I/O
  PID USER     PRI  NI  VIRT   RES   SHR  S  CPU% MEM%   TIME+  Command
774915 root      20   0  224M  4152  3432  R   0.6  0.5   0:00.32 htop
   1 root      20   0  241M  5916  3700  S   0.0  0.8   8:14.16 /usr/lib/systemd/systemd --switched-root --system --des
  451 root      20   0  114M  6404  4920  S   0.0  0.8   7:00.55 /usr/lib/systemd/systemd-journald
  506 root      16  -4  127M   804   548  S   0.0  0.1   3:09.61 /sbin/auditd
  507 root      16  -4  127M   804   548  S   0.0  0.1   0:05.24 /sbin/auditd
  508 root      16  -4 48724    64     0  S   0.0  0.0   1:01.64 /usr/sbin/sedispatch
  509 root      16  -4  127M   804   548  S   0.0  0.1   0:33.86 /sbin/auditd
  531 root      20   0 97108   544   472  S   0.0  0.1   0:03.40 /usr/lib/systemd/systemd-udev
  544 root      20   0  6764   288   276  S   0.0  0.0   0:00.00 /usr/sbin/mcelog --ignoreudev --daemon --foreground
  551 libstorage 20   0  8688   320   268  S   0.0  0.0   0:04.15 /usr/bin/lsm -d
  555 root      20   0 50484    96     0  S   0.0  0.0   0:00.34 /usr/sbin/smartd -n -q never
  556 daemon    20   0  157M     0     0  S   0.0  0.0   0:00.00 /usr/sbin/rngd -f --fill-watermark=0 -x pkcs11 -x nist
  560 root      20   0 81912  1564  1176  S   0.0  0.2   1:07.96 /usr/lib/systemd/systemd-logind
  562 dbus      20   0 68148  1724   952  S   0.0  0.2   6:05.77 /usr/bin/dbus-daemon --system --address=systemd: --nofo
  567 root      20   0  658M   820   820  S   0.0  0.1   0:04.01 /usr/libexec/platform-python -s /usr/sbin/firewalld --n
  571 chrony   20   0  136M   988   792  S   0.0  0.1   0:05.17 /usr/sbin/chronyd
  608 root      20   0  576M   964     0  S   0.0  0.1   0:49.72 /usr/sbin/NetworkManager --no-daemon
  614 root      20   0  576M   964     0  S   0.0  0.1   0:26.93 /usr/sbin/NetworkManager --no-daemon
  615 root      20   0  576M   964     0  S   0.0  0.1   0:09.35 /usr/sbin/NetworkManager --no-daemon
F1Help F2Setup F3Search F4Filter F5Tree F6SortBy F7Nice F8Nice F9Kill F10Quit

```

- **ps aux** — Show all running processes.

```

[root@srv ~]# ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.7 246976 5916 ?        Ss   Oct24   8:14 /usr/lib/systemd/systemd --switched-root --system --des
root         2  0.0  0.0     0     0 ?        S    Oct24   0:00 [kthreadd]
root         3  0.0  0.0     0     0 ?        I<   Oct24   0:00 [rcu_gp]
root         4  0.0  0.0     0     0 ?        I<   Oct24   0:00 [rcu_par_gp]
root         5  0.0  0.0     0     0 ?        I<   Oct24   0:00 [slub_flushwq]
root         7  0.0  0.0     0     0 ?        I<   Oct24   0:00 [kworker/0:0H-events_highpri]
root         9  0.0  0.0     0     0 ?        I    Oct24   0:22 [kworker/u2:0-events_unbound]
root        10  0.0  0.0     0     0 ?        I<   Oct24   0:00 [mm_percpu_wq]
root        11  0.0  0.0     0     0 ?        S    Oct24   0:00 [rcu tasks rude ]

```

- **kill [PID]** — Terminate a process by its Process ID (PID).
- **Service Management:**
 - **systemctl start/stop/restart [service]** — Start, stop, or restart a service.
 - **systemctl status [service]** — Check the status of a service.

```
[root@srv ~]# systemctl stop mariadb.service
[root@srv ~]# systemctl status mariadb.service
● mariadb.service - MariaDB 10.11.10 database server
   Loaded: loaded (/usr/lib/systemd/system/./mariadb.service; enabled; vendor preset: disabled)
   Drop-In: /etc/systemd/system/mariadb.service.d
            └─migrated-from-mv.cnf-settings.conf
   Active: inactive (dead) since Tue 2024-11-12 01:43:06 EST; 13s ago
     Docs: man:mariadb(8)
           https://mariadb.com/kb/en/library/systemd/
   Process: 462335 ExecStart=/usr/sbin/mariadb $MYSQLD_OPTS $WSREP_NEW_CLUSTER $WSREP_START_POSITION (code=exited, status=0/SUCCESS)
  Main PID: 462335 (code=exited, status=0/SUCCESS)
   Status: "MariaDB server is down"

Nov 12 01:43:04 srv.lintsawa.com mariadb[462335]: 2024-11-12 1:43:04 0 [Note] /usr/sbin/mariadb (initiated by: unknown)
Nov 12 01:43:04 srv.lintsawa.com mariadb[462335]: 2024-11-12 1:43:04 0 [Note] InnoDB: FTS optimize thread exiting.
Nov 12 01:43:04 srv.lintsawa.com mariadb[462335]: 2024-11-12 1:43:04 0 [Note] InnoDB: Starting shutdown...
Nov 12 01:43:04 srv.lintsawa.com mariadb[462335]: 2024-11-12 1:43:04 0 [Note] InnoDB: Dumping buffer pool(s) to /var/lib
Nov 12 01:43:04 srv.lintsawa.com mariadb[462335]: 2024-11-12 1:43:04 0 [Note] InnoDB: Buffer pool(s) dump completed at
Nov 12 01:43:06 srv.lintsawa.com mariadb[462335]: 2024-11-12 1:43:06 0 [Note] InnoDB: Removed temporary tablespace data
Nov 12 01:43:06 srv.lintsawa.com mariadb[462335]: 2024-11-12 1:43:06 0 [Note] InnoDB: Shutdown completed; log sequence
Nov 12 01:43:06 srv.lintsawa.com mariadb[462335]: 2024-11-12 1:43:06 0 [Note] /usr/sbin/mariadb: Shutdown complete
Nov 12 01:43:06 srv.lintsawa.com systemd[1]: mariadb.service: Succeeded.
Nov 12 01:43:06 srv.lintsawa.com systemd[1]: Stopped MariaDB 10.11.10 database server.
[root@srv ~]# systemctl start mariadb.service
● mariadb.service - MariaDB 10.11.10 database server
   Loaded: loaded (/usr/lib/systemd/system/./mariadb.service; enabled; vendor preset: disabled)
   Drop-In: /etc/systemd/system/mariadb.service.d
            └─migrated-from-mv.cnf-settings.conf
   Active: active (running) since Tue 2024-11-12 01:45:44 EST; 2s ago
```

Familiarity with these commands will enable you to manage the server effectively, troubleshoot issues, and control resources.

3. Monitoring System Resources (CPU, Memory, Disk Usage)

Monitoring system resources ensures the server is running efficiently and prevents potential performance issues.

Monitoring in CWP:

- This can be seen on the CWP Admin Panel dashboard.

Key Monitoring Tools via SSH.

1. **CPU Usage:**
 - **top** and **htop** (if installed) provide real-time CPU usage for processes.
 - Monitor CPU load; if consistently high, it could indicate resource-heavy applications or attacks.
2. **Memory Usage:**

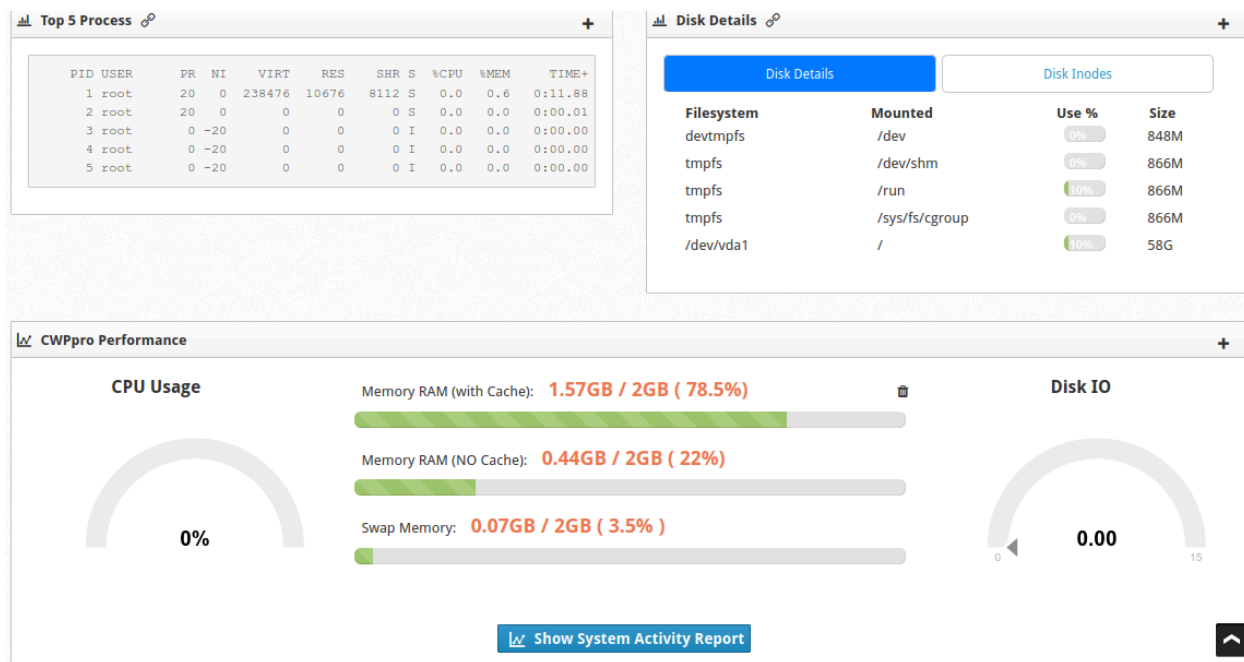
- **free -m** shows current memory usage, including free and available memory.
- Excessive memory usage may suggest memory leaks or resource-intensive processes.

3. Disk Usage:

- **df -h** displays disk space usage for mounted file systems.
- **du -sh /path/to/directory** shows disk usage of specific directories.
- Regularly check disk usage to ensure sufficient space for applications and data storage.

4. Resource Graphs in CWP:

- CWP provides graphical representations of CPU, memory, and disk usage over time, making it easy to track resource trends and anticipate needs.



4. Scheduled Tasks and Cron Jobs

Cron jobs allow you to automate repetitive tasks, such as backups, updates, or cleanup operations, at specific intervals.

Setting Up Cron Jobs in CWP:

1. Access Cron Jobs:

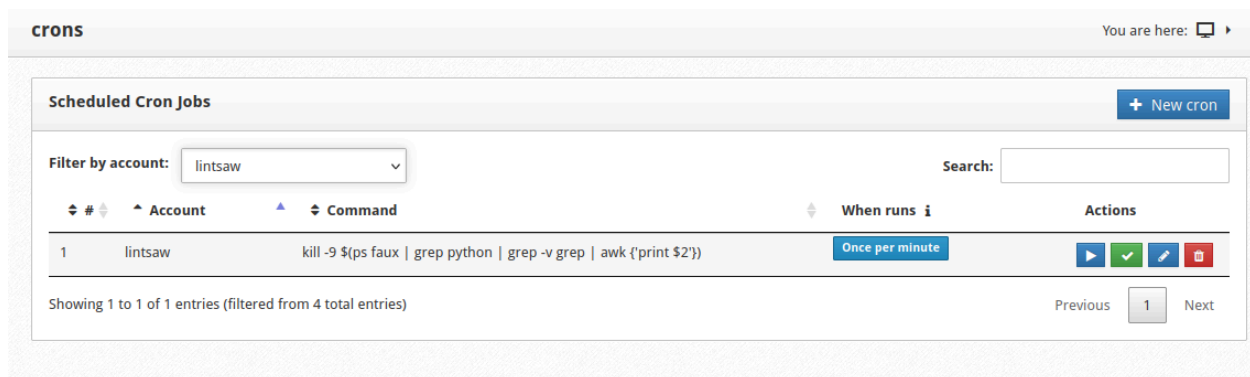
- In the CWP Admin Panel, go to **Server Settings > Crontab for Users/root**

2. Creating a Cron Job:

- Specify the command to run and the frequency (e.g., every minute, hour, day).
- Use the format: **Minute Hour Day Month Day-of-week Command**.
 - Example: **0 3 * * * /usr/bin/backup.sh** (runs daily at 3 AM).

Example

The cron in this example runs every minute and kills python processes.



The screenshot shows the 'crons' management interface. At the top, it says 'Scheduled Cron Jobs' with a '+ New cron' button. Below this, there's a 'Filter by account:' dropdown set to 'lintsaw' and a 'Search:' input field. A table lists the cron jobs with columns for '#', 'Account', 'Command', 'When runs', and 'Actions'. One job is visible: #1, Account: lintsaw, Command: kill -9 \$(ps faux | grep python | grep -v grep | awk {'print \$2'}), When runs: Once per minute. The 'Actions' column contains icons for play, check, edit, and delete. At the bottom, it says 'Showing 1 to 1 of 1 entries (filtered from 4 total entries)' and has 'Previous', '1', and 'Next' navigation buttons.

3. Basic Cron Syntax:

- ***** — Runs the command at every interval (e.g., every minute, hour).
- Numbers or ranges can specify exact intervals (e.g., **0 3 * *** for 3 AM daily).
- Example Commands:
 - Backup: **mysqldump -u username -p database_name > /path/to/backup.sql**
 - Disk Cleanup: **rm -rf /tmp/*** (clears temporary files).

4. Viewing and Managing Cron Jobs:

- List active cron jobs for the current user with **crontab -l**.

```
[root@srv2 ~]# crontab -l
00 03 * * * /usr/local/cwp/php71/bin/php -d max_execution_time=18000 -q /usr/local/cwpsrv/htdocs/resources/admin/include/cron_autossl_all_domains.php
0 0 * * * /usr/local/cwp/php71/bin/php -d max_execution_time=18000 -q /usr/local/cwpsrv/htdocs/resources/admin/include/alertandautorenewssl.php
26 1 * * * "/root/.acme.sh"/acme.sh --cron --home "/root/.acme.sh/cwp_certs" > /dev/null
```

- Add or edit cron jobs with **crontab -e**.