

CloudLinux is a Linux operating system optimized for service providers. It is commercially supported and improves density, stability, and security in multitenant environments such as shared hosting. It is easy to deploy because it is interchangeable with the most popular RPM-based Linux distributions on the market.

Industry awards



CloudLinux employs a wide range of the latest technologies and a custom Linux kernel to achieve a multitude of benefits, including the following:



LVE Manager



Ruby Selector



CageFS



Python Selector



MySQL Governor



Mod Isapi



PHP Selector



OptimumCache





Today, a single site can consume all CPU, memory, and IO resources—and bring the server to a halt. LVE (lightweight virtual environment) technology prevents that by allowing admin to set up individual resource limits. This ensures that a tenant can never use more resources than he or she is given.

LVE is a kernel-level technology developed by the CloudLinux team. It integrates at the server, PAM (pluggable authentication modules), and database levels to prevent any kind of abuse while maintaining the lowest overhead possible. The technology has roots in common with container-based virtualization.

LVE allows the host to maintain fine-tuned control over resources—including CPU, IO, memory, inodes, number of processes, and connections—that any single account can use. It is lightweight and transparent.

The goal of LVE is to ensure that no single Web site can bring down

Memory limits

Memory limits control the amount of memory each customer can use. CloudLinux is able to identify, in real time, the amount of memory actually used by an end customer's processes. Physical memory limits are especially effective in preventing out of memory (OOM) issues and customers' ballooning memory usage, which destroy caches and cause server overload.

IO limits

IO limits restrict the data throughput for the customer. They are measured in KB/s. When the limit is reached, the processes are throttled (put to sleep). Because IO is one of the scarcest resources in shared hosting, the ability to put an upper limit on customer use is vital.

CPU limits

CPU limits establish the maximum amount of CPU resources that an account can use. When a user hits the CPU limit, processes within that limit are slowed down. CPU limits are crucial in preventing CPU usage spikes, which can often make servers slow and unresponsive.

Number-of-processes limits

Number-of-processes limits control the total number of processes within LVE. Once the limit is reached, no new process can be created until another one finished. This effectively prevents fork bombs and similar DoS attacks.

Entry processes limits

Entry processes limits control the number of entries into LVE. The best way to think about this type of limit is as the number of Web scripts that can be executed in parallel by visitors to a site. These limits are important to preventing single sites from hogging all Apache slots, thus causing Apache to be unresponsive.

Inodes limits

An inode is a data structure on a file system that is used to keep information about a file or a folder. The number of inodes indicates the number of files and folders an account has. Inodes limit work on the level of disk quota.





CageFS

CageFS is a virtualized per-user file system that uniquely encapsulates each customer, preventing users from seeing one another and viewing sensitive information about the server. CageFS prevents a large number of attacks, including most privilege-escalation and information-disclosure attacks. The best part is that CageFS is completely transparent to customers and does not require them to change their scripts.

CageFS Benefits:

- Only safe binaries are available to users.
- Users do not see other users and have no way to detect the presence of other users or their user names on the server.
- Users cannot see server configuration files such as Apache config files.
- Users have a limited view of their own processing file system and cannot see other users' processes.

At the same time, a user's environment is fully functional, and he or she should not feel restricted in any way. No adjustments to user scripts are necessary.



MySQL Governor

MySQL often becomes a major headache for shared hosting companies. Keeping MySQL stable might be difficult, and customer queries can easily slow everything down. This is where MySQL Governor comes in. Its ability to pinpoint abusers and throttle them in real time is unprecedented in the industry. With support from the latest versions of MySQL and MariaDB, it is a must-have for any shared host.

MySQL Governor tracks CPU and disk IO usage for every user in real time and throttles MySQL queries by using same-per-user LVE limits. By using the dbtop utility, it is possible to see usage as it happens on a per-customer basis, ensuring that system admins always know what is going on.

MySQL Governor Benefits:

- Stable, simple limits counted as part of LVE limits
- Automatic detection and killing of long-running queries
- MySQL 5.1–5.6 support
- MariaDB 5.x and 10.x support

As a result, hosting companies can carefully throttle resources to maintain performance across servers without instantly disconnecting users.

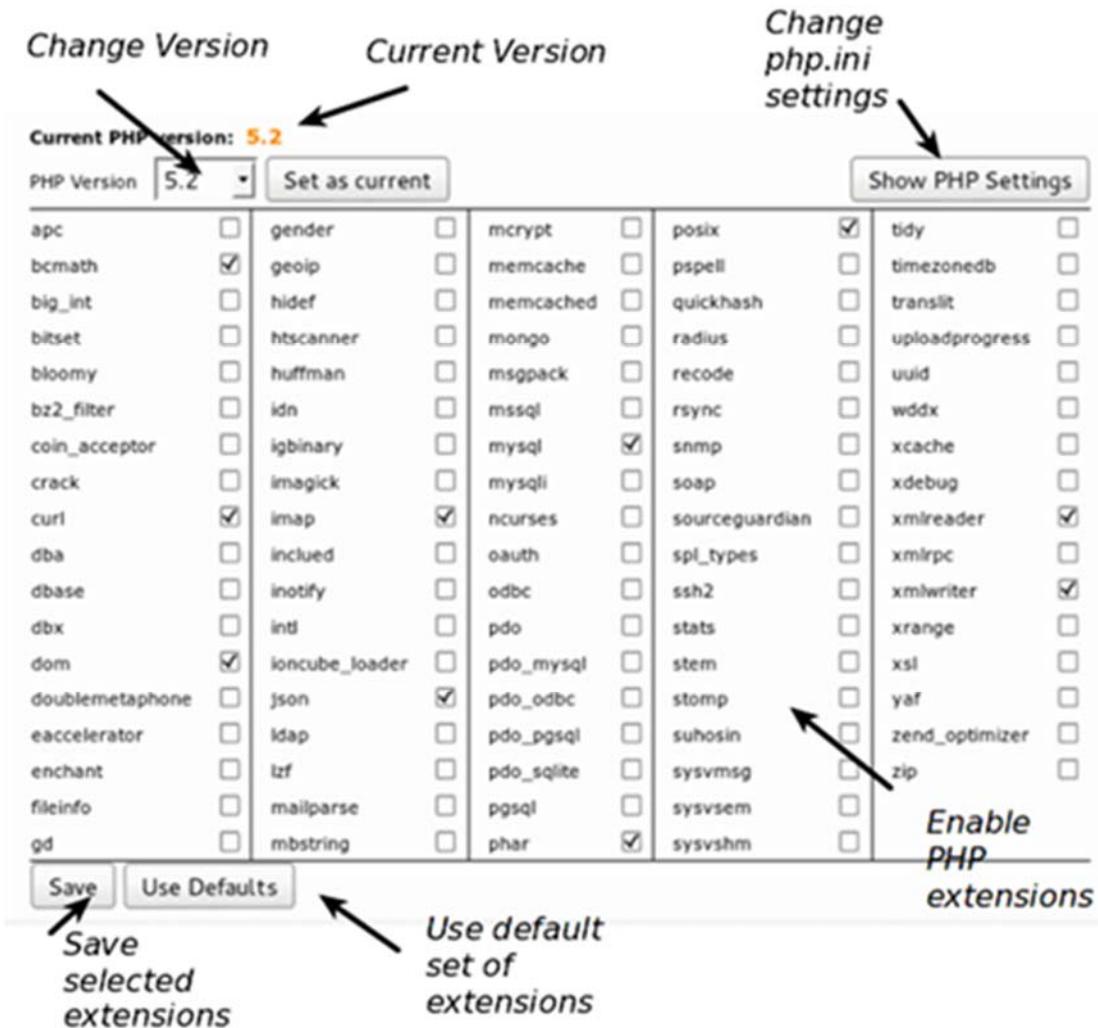




The PHP Selector is a part of CloudLinux that allows end users to select the specific version of PHP they need. It allows ultimate flexibility by offering all popular versions of PHP, with more than 120 PHP extensions to choose from.

CloudLinux packages PHP versions 4.4, 5.1, 5.2, 5.3, 5.4, 5.5, and 5.6. The convenient UI lets a customer switch between versions, select the extensions, and adjust PHP settings.

No longer do you have to move customers to other servers or force them to switch to VPS (or another hosting company) just to get that old PHP 5.2 that their scripts require.



Change Version (points to PHP Version dropdown)

Current Version (points to 'Current PHP version: 5.2')

Change php.ini settings (points to 'Show PHP Settings' button)

Current PHP version: **5.2**

PHP Version: **5.2** | Set as current

apc	<input type="checkbox"/>	gender	<input type="checkbox"/>	mcrypt	<input type="checkbox"/>	posix	<input checked="" type="checkbox"/>	tidy	<input type="checkbox"/>
bcmath	<input checked="" type="checkbox"/>	geopip	<input type="checkbox"/>	memcache	<input type="checkbox"/>	pspell	<input type="checkbox"/>	timezonedb	<input type="checkbox"/>
big_int	<input type="checkbox"/>	hidef	<input type="checkbox"/>	memcached	<input type="checkbox"/>	quickhash	<input type="checkbox"/>	translit	<input type="checkbox"/>
bitset	<input type="checkbox"/>	htscanner	<input type="checkbox"/>	mongo	<input type="checkbox"/>	radius	<input type="checkbox"/>	uploadprogress	<input type="checkbox"/>
bloomy	<input type="checkbox"/>	huffman	<input type="checkbox"/>	msgpack	<input type="checkbox"/>	recode	<input type="checkbox"/>	uuid	<input type="checkbox"/>
bz2_filter	<input type="checkbox"/>	idn	<input type="checkbox"/>	mssql	<input type="checkbox"/>	rsync	<input type="checkbox"/>	wddx	<input type="checkbox"/>
coin_acceptor	<input type="checkbox"/>	igbinary	<input type="checkbox"/>	mysql	<input checked="" type="checkbox"/>	snmp	<input type="checkbox"/>	xcache	<input type="checkbox"/>
crack	<input type="checkbox"/>	imagick	<input type="checkbox"/>	mysqli	<input type="checkbox"/>	soap	<input type="checkbox"/>	xdebug	<input type="checkbox"/>
curl	<input checked="" type="checkbox"/>	imap	<input checked="" type="checkbox"/>	ncurses	<input type="checkbox"/>	sourceguardian	<input type="checkbox"/>	xmlreader	<input checked="" type="checkbox"/>
dba	<input type="checkbox"/>	includ	<input type="checkbox"/>	oauth	<input type="checkbox"/>	spl_types	<input type="checkbox"/>	xmlrpc	<input type="checkbox"/>
dbase	<input type="checkbox"/>	inotify	<input type="checkbox"/>	odbc	<input type="checkbox"/>	ssh2	<input type="checkbox"/>	xmlwriter	<input checked="" type="checkbox"/>
dbx	<input type="checkbox"/>	intl	<input type="checkbox"/>	pdo	<input type="checkbox"/>	stats	<input type="checkbox"/>	xrange	<input type="checkbox"/>
dom	<input checked="" type="checkbox"/>	ioncube_loader	<input type="checkbox"/>	pdo_mysql	<input type="checkbox"/>	stem	<input type="checkbox"/>	xsl	<input type="checkbox"/>
doublemetaphone	<input type="checkbox"/>	json	<input checked="" type="checkbox"/>	pdo_odbc	<input type="checkbox"/>	stomp	<input type="checkbox"/>	yaf	<input type="checkbox"/>
eaccelerator	<input type="checkbox"/>	ldap	<input type="checkbox"/>	pdo_pgsql	<input type="checkbox"/>	suhosin	<input type="checkbox"/>	zend_optimizer	<input type="checkbox"/>
enchant	<input type="checkbox"/>	lzf	<input type="checkbox"/>	pdo_sqlite	<input type="checkbox"/>	sysvmsg	<input type="checkbox"/>	zip	<input type="checkbox"/>
fileinfo	<input type="checkbox"/>	mailparse	<input type="checkbox"/>	pgsql	<input type="checkbox"/>	sysvsem	<input type="checkbox"/>		
gd	<input type="checkbox"/>	mbstring	<input type="checkbox"/>	phar	<input checked="" type="checkbox"/>	sysvshm	<input type="checkbox"/>		

Save | Use Defaults

Enable PHP extensions (points to the extension list)

Save selected extensions (points to 'Save' button)

Use default set of extensions (points to 'Use Defaults' button)

CloudLinux includes a large number of extensions and gives control to the end user. Only users who need non-default version of PHP will be using PHP selector. Everyone else will use the default PHP version installed on the server, so there is no extra work associated with the PHP selector.





Ruby Selector

The Ruby Selector allows end users to choose the Ruby version for application and to install additional modules (gems) to the application environment. Ruby Selector uses mod_passenger for optimum performance.

Ruby Selector Benefits:

- Supports 1.8, 1.9, 2.0, and 2.1 Ruby versions
- Allows users to install additional modules to application environments



Python Selector

Python Selector allows end users to choose the Python version as an application and to install additional modules. Python Selector uses mod_passenger to get the best performance from Python applications.



Mod_Isapi

Mod_Isapi is the fastest and most reliable way to serve PHP pages. It is a drop-in replacement for SuPHP, FCGID, RUID2, and ITK. It has a low memory footprint and understands PHP directives from .htaccess files.

Benefits of Mod_Isapi compared to other ways to serve PHP:

- Faster than any other way to serve PHP with Apache
- Doesn't suffer from stability issues in process management like PHP FPM and mod_fcgid
- Full benefits of opcode caching
- Compatible with MPM Worker & Event
- No tuning required
- Support for PHP directives in .htaccess files
- Drop-in replacement for existing ways to serve PHP
- Fully compatible with PHP Selector



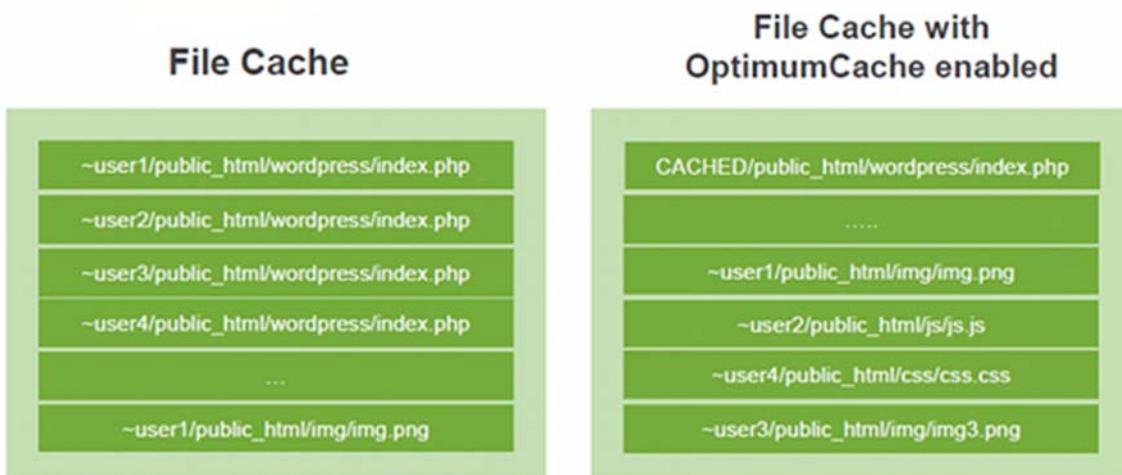


Today, a typical shared hosting server encompasses multiple Web sites that are using WordPress, Joomla, and other popular software. This software is composed of exactly the same files, yet the files have to be separately loaded for each Web site, wasting both server IO and memory. OptimumCache solves these problems by creating a cache of duplicate files.

OptimumCache is a component that handles duplicate files so that they are loaded just once from the file system. In doing so, the system bypasses disk IO, significantly improving the speed of reading duplicate files while lowering load on the hard disk.

OptimumCache software can double IO throughput on a typical shared-hosting system by eliminating IO operations related to reading duplicate files, thus optimizing file caching.

- OptimumCache creates a cache of duplicate files (WordPress, Joomla, and other popular software).
- Files are loaded from the file system cache.
- System bypasses disk IO, significantly improving the speed of reading files while lowering load on the hard disk.



With OptimumCache, if a duplicate of a previously loaded file is requested, the file gets loaded from the file system cache. OptimumCache reads and stores the file just once—minimizing the number of duplicates in the file system cache—and improves overall cache efficiency. In turn, it reduces memory usage and decreases the number of disk operations—while improving the Web site’s response time.





IGAL ROSENKRANZ
CEO
LIVEDNS LTD

As a hosting company we are very impressed with CloudLinux product. It's deployed across all our cPanel shared hosting servers and now we can provide more stable and secured service to our customers.

CloudLinux has been a key component to Funio's success since day one. Combined with CageFS, it has helped us reach new heights in terms of server stability by preventing the "bad neighbor effect" while increasing overall server security and density. Installing CloudLinux on a shared-hosting server instantly makes it more reliable and more secure without having to invest much effort, and I wouldn't build a shared-hosting server without it!

Funio.

JEAN-FRANCOIS CHEVRETTE
TECH LEAD
FUNIO



ERNEST BRUGGEMAN
CEO
EBRIDGE ICT

We're extremely satisfied with both the CloudLinux product and their level of support. Our clients really benefit from the improved reliability of our hosting services, our job of identifying and taking care of troublesome accounts has become much easier, and the CloudLinux customer support really sets the bar! Highly recommended!

Our shared hosting division at HostDime was always in need of a solution like CloudLinux but nothing of it's kind existed. We met Igor in person at HostingCon 2010 and now, thanks to this amazing product and superb support from Igor, we power hundreds of servers with it.



FILIFE MENDES
PRESIDENT
HOSTDIME INC



MySQL GOVERNOR



PHP SELECTOR



HARDENED KERNEL



TENANT ISOLATION



24/7 SUPPORT



FAST UPDATES



RESOURCE LIMITATION