

Puma英文文档

书栈(BookStack.CN)

目 录

[致谢](#)

[README](#)

[Docs](#)

[Overview](#)

[Deployment engineering for puma](#)

[Nginx configuration example file](#)

[Plugins](#)

[Restarts](#)

[Sending Signals](#)

[Service Configuration](#)

[Tools](#)

[Puma daemon service](#)

[Puma as a service using rc.d](#)

[Puma as a service](#)

[Puma as a service using Upstart](#)

[HISTORY\(升级日志\)](#)

[Release](#)

致谢

当前文档 《Puma英文文档》 由 进击的皇虫 使用 书栈(BookStack.CN) 进行构建,生成于 2018-07-15。

书栈(BookStack.CN) 仅提供文档编写、整理、归类等功能,以及对文档内容的生成和导出工具。

文档内容由网友们编写和整理,书栈(BookStack.CN) 难以确认文档内容知识点是否错漏。如果您在阅读文档获取知识的时候,发现文档内容有不恰当的地方,请向我们反馈,让我们共同携手,将知识准确、高效且有效地传递给每一个人。

同时,如果您在日常工作、生活和学习中遇到有价值有营养的知识文档,欢迎分享到 书栈(BookStack.CN) ,为知识的传承献上您的一份力量!

如果当前文档生成时间太久,请到 书栈(BookStack.CN) 获取最新的文档,以跟上知识更新换代的步伐。

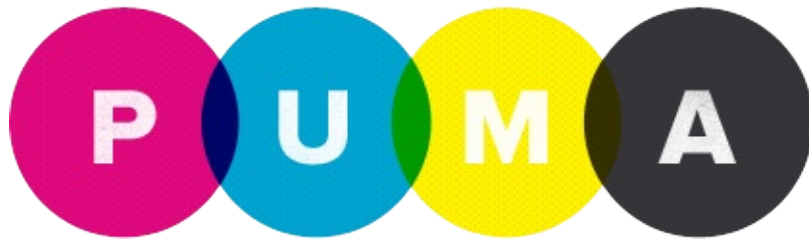
文档地址: <http://www.bookstack.cn/books/puma-en>

书栈官网: <http://www.bookstack.cn>

书栈开源: <https://github.com/TruthHun>

分享,让知识传承更久远! 感谢知识的创造者,感谢知识的分享者,也感谢每一位阅读到此处的读者,因为我们都将成为知识的传承者。

README



Puma: A Ruby Web Server Built For Concurrency

[gitter](#) [join chat](#)

build failing

build passing

FIXME Migrate to GitLab

⬆ maintainability D

Puma is a **simple, fast, threaded, and highly concurrent HTTP 1.1 server for Ruby/Rack applications** in development and production.

Built For Speed & Concurrency

Under the hood, Puma processes requests using a C-optimized Rake1 extension (inherited from Mongrel) that provides fast, accurate HTTP 1.1 protocol parsing in a portable way. Puma then serves the request in a thread from an internal thread pool. Since each request is served in a separate thread, truly concurrent Ruby implementations (JRuby, Rubinius) will use all available CPU cores.

Puma was designed to be the go-to server for [Rubinius](#), but also works well with JRuby and MRI.

On MRI, there is a Global VM Lock (GVL) that ensures only one thread can run Ruby code at a time. But if you're doing a lot of blocking IO (such as HTTP calls to external APIs like Twitter), Puma still improves MRI's throughput by allowing blocking IO to be run concurrently.

Quick Start

```
1. $ gem install puma
```

```
2. $ puma <any rackup (*.ru) file>
```

Frameworks

Rails

Puma is the default server for Rails, and should already be included in your Gemfile.

Then start your server with the `rails` command:

```
1. $ rails s
```

Many configuration options are not available when using `rails s`. It is recommended that you use Puma's executable instead:

```
1. $ bundle exec puma
```

Sinatra

You can run your Sinatra application with Puma from the command line like this:

```
1. $ ruby app.rb -s Puma
```

Or you can configure your application to always use Puma:

```
1. require 'sinatra'
2. configure { set :server, :puma }
```

Configuration

Puma provides numerous options. Consult `puma -h` (or `puma --help`) for a full list of CLI options, or see [dsl.rb](#).

Thread Pool

Puma uses a thread pool. You can set the minimum and maximum number of threads that are available in the pool with the `-t` (or `--threads`) flag:

```
1. $ puma -t 8:32
```

Puma will automatically scale the number of threads, from the minimum until it caps out at the maximum, based on how much traffic is present. The current default is `0:16`. Feel free to experiment, but be careful not to set the number of maximum

threads to a large number, as you may exhaust resources on the system (or hit resource limits).

Be aware that additionally Puma creates threads on its own for internal purposes (e.g. handling slow clients). So even if you specify `-t 1:1`, expect around 7 threads created in your application.

Clustered mode

Puma also offers “clustered mode”. Clustered mode `fork`s workers from a master process. Each child process still has its own thread pool. You can tune the number of workers with the `-w` (or `--workers`) flag:

```
1. $ puma -t 8:32 -w 3
```

Note that threads are still used in clustered mode, and the `-t` thread flag setting is per worker, so `-w 2 -t 16:16` will spawn 32 threads in total.

In clustered mode, Puma may “preload” your application. This loads all the application code *prior* to forking. Preloading reduces total memory usage of your application via an operating system feature called [copy-on-write](#) (Ruby 2.0+ only). Use the `--preload` flag from the command line:

```
1. $ puma -w 3 --preload
```

If you’re using a configuration file, use the `preload_app!` method:

```
1. # config/puma.rb
2. workers 3
3. preload_app!
```

Additionally, you can specify a block in your configuration file that will be run on boot of each worker:

```
1. # config/puma.rb
2. on_worker_boot do
3.   # configuration here
4. end
```

This code can be used to setup the process before booting the application, allowing you to do some Puma-specific things that you don’t want to embed in your application. For instance, you could fire a log notification that a worker booted or send something to statsd.

This can be called multiple times.

If you’re preloading your application and using ActiveRecord, it’s recommended that you setup your connection pool here:

```

1. # config/puma.rb
2. on_worker_boot do
3.   ActiveSupport.on_load(:active_record) do
4.     ActiveRecord::Base.establish_connection
5.   end
6. end

```

On top of that, you can specify a block in your configuration file that will be run before workers are forked:

```

1. # config/puma.rb
2. before_fork do
3.   # configuration here
4. end

```

Preloading can't be used with phased restart, since phased restart kills and restarts workers one-by-one, and `preload_app` copies the code of master into the workers.

Binding TCP / Sockets

In contrast to many other server configs which require multiple flags, Puma simply uses one URI parameter with the `-b` (or `--bind`) flag:

```

1. $ puma -b tcp://127.0.0.1:9292

```

Want to use UNIX Sockets instead of TCP (which can provide a 5-10% performance boost)?

```

1. $ puma -b unix:///var/run/puma.sock

```

If you need to change the permissions of the UNIX socket, just add a `umask` parameter:

```

1. $ puma -b 'unix:///var/run/puma.sock?umask=0111'

```

Need a bit of security? Use SSL sockets:

```

1. $ puma -b 'ssl://127.0.0.1:9292?key=path_to_key&cert=path_to_cert'

```

Controlling SSL Cipher Suites

Need to use or avoid specific SSL cipher suites? Use `ssl_cipher_filter` or `ssl_cipher_list` options.

Ruby:

```

1. $ puma -b 'ssl://127.0.0.1:9292?key=path_to_key&cert=path_to_cert&ssl_cipher_filter=!aNULL:AES+SHA'

```

JRuby:

```
1. $ puma -b 'ssl://127.0.0.1:9292?keystore=path_to_keystore&keystore-pass=keystore_password&ssl_cipher_list=TLS_RSA_WITH_AES_128_CBC_SHA,TLS_RSA_WITH_AES_256_CBC_SHA'
```

See <https://www.openssl.org/docs/man1.0.2/apps/ciphers.html> for cipher filter format and full list of cipher suites.

Control/Status Server

Puma has a built-in status/control app that can be used to query and control Puma itself.

```
1. $ puma --control-url tcp://127.0.0.1:9293 --control-token foo
```

Puma will start the control server on localhost port 9293. All requests to the control server will need to include `token=foo` as a query parameter. This allows for simple authentication. Check out [status.rb](#) to see what the app has available.

You can also interact with the control server via `pumactl`. This command will restart Puma:

```
1. $ pumactl --control-url 'tcp://127.0.0.1:9293' --control-token foo restart
```

To see a list of `pumactl` options, use `pumactl --help`.

Configuration File

You can also provide a configuration file which Puma will use with the `-C` (or `--config`) flag:

```
1. $ puma -C /path/to/config
```

If no configuration file is specified, Puma will look for a configuration file at `config/puma.rb`. If an environment is specified, either via the `-e` and `--environment` flags, or through the `RACK_ENV` environment variable, the default file location will be `config/puma/environment_name.rb`.

If you want to prevent Puma from looking for a configuration file in those locations, provide a dash as the argument to the `-C` (or `--config`) flag:

```
1. $ puma -C "-"
```

Take the following [sample configuration](#) as inspiration or check out [dsl.rb](#) to see all available options.

Restart

Puma includes the ability to restart itself. When available (MRI, Rubinius, JRuby), Puma performs a “hot restart”. This is the same functionality available in *Unicorn* and *NGINX* which keep the server sockets open between restarts. This makes sure that no pending requests are dropped while the restart is taking place.

For more, see the [restart documentation](#).

Signals

Puma responds to several signals. A detailed guide to using UNIX signals with Puma can be found in the [signals documentation](#).

Platform Constraints

Some platforms do not support all Puma features.

- **JRuby, Windows:** server sockets are not seamless on restart, they must be closed and reopened. These platforms have no way to pass descriptors into a new process that is exposed to Ruby. Also, cluster mode is not supported due to a lack of `fork(2)`.
- **Windows:** daemon mode is not supported due to a lack of `fork(2)`.

Known Bugs

For MRI versions 2.2.7, 2.2.8, 2.2.9, 2.2.10 2.3.4 and 2.4.1, you may see `stream closed in another thread (IOError)`. It may be caused by a [Ruby bug](#). It can be fixed with the gem https://rubygems.org/gems/stopgap_13632:

```
1. if %w(2.2.7 2.2.8 2.2.9 2.2.10 2.3.4 2.4.1).include? RUBY_VERSION
2.   begin
3.     require 'stopgap_13632'
4.   rescue LoadError
5.   end
6. end
```

Deployment

Puma has support for Capistrano with an [external gem](#).

It is common to use process monitors with Puma. Modern process monitors like `systemd` or `upstart` provide continuous monitoring and restarts for increased reliability in production environments:

- [tools/jungle](#) for sysvinit (init.d) and upstart
- [docs/systemd](#)

Contributing

To run the test suite:

1. `$ bundle install`
2. `$ bundle exec rake`

License

Puma is copyright Evan Phoenix and contributors, licensed under the BSD 3-Clause license. See the included LICENSE file for details.

来源(书栈小编注)

<https://github.com/puma/puma>

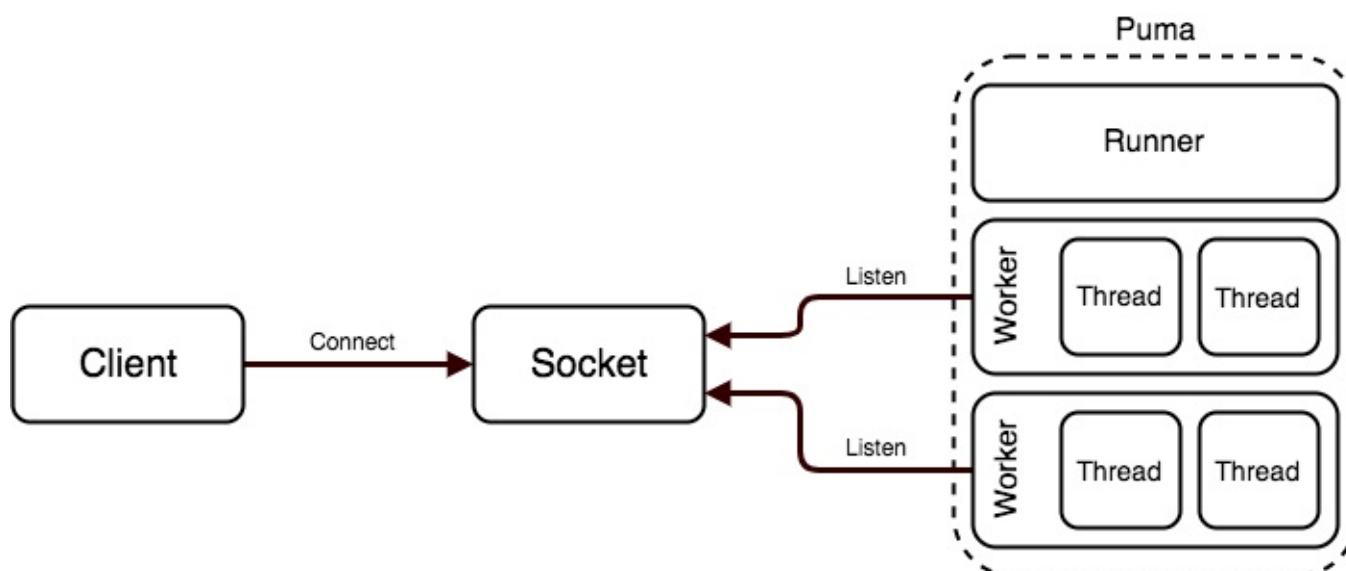
Docs

- [Overview](#)
- [Deployment engineering for puma](#)
- [Nginx configuration example file](#)
- [Plugins](#)
- [Restarts](#)
- [Sending Signals](#)
- [Service Configuration](#)

Overview

Architecture

Overview

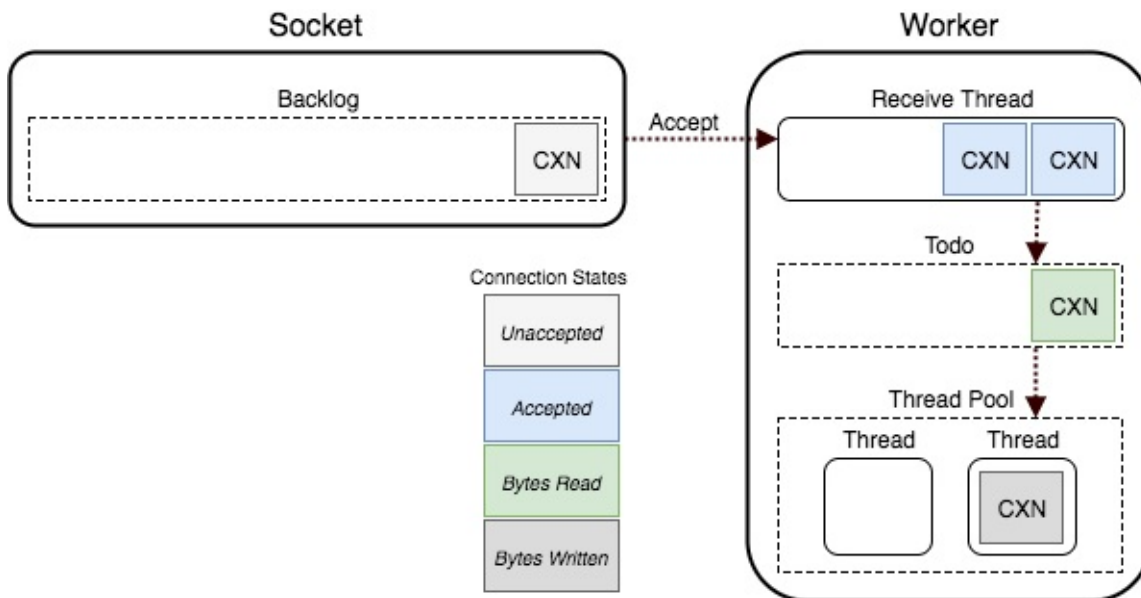


Puma is a threaded web server, processing requests across a TCP or UNIX socket.

Workers accept connections from the socket and a thread in the worker's thread pool processes the client's request.

Clustered mode is shown/discussed here. Single mode is analogous to having a single worker process.

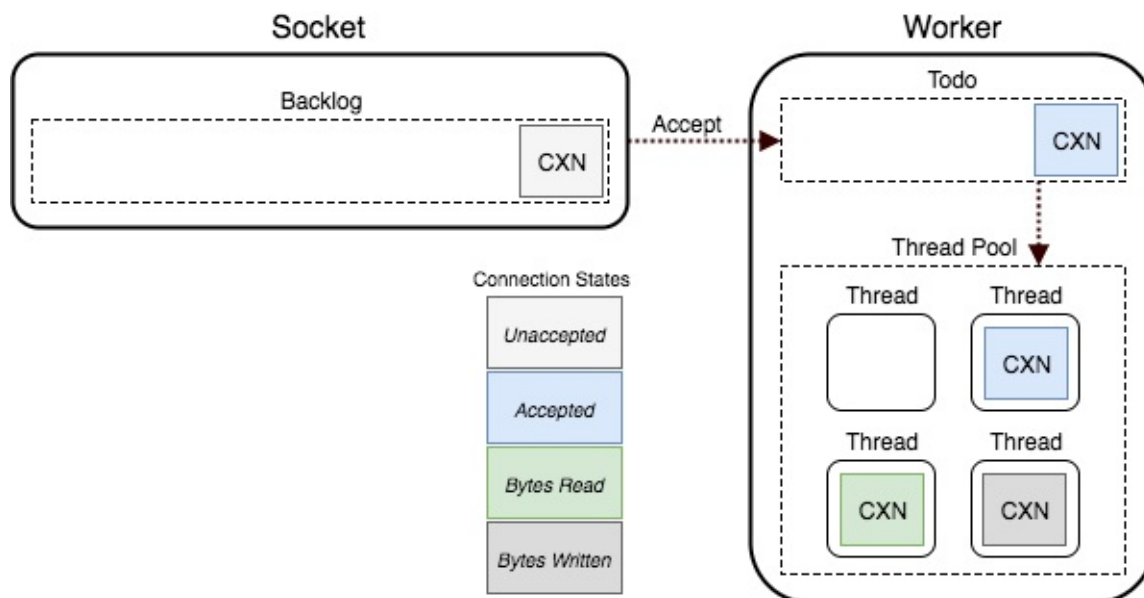
Connection pipeline



- Upon startup, Puma listens on a TCP or UNIX socket.
 - The backlog of this socket is configured (with a default of 1024), determining how many established but unaccepted connections can exist concurrently.
 - This socket backlog is distinct from the “backlog” of work as reported by the control server stats. The latter is the number of connections in that worker’s “todo” set waiting for a worker thread.
- By default, a single, separate thread is used to receive HTTP requests across the socket.
 - When at least one worker thread is available for work, a connection is accepted and placed in this request buffer
 - This thread waits for entire HTTP requests to be received over the connection
 - Once received, the connection is pushed into the “todo” set
- Worker threads pop work off the “todo” set for processing
 - The thread processes the request via the rack application (which generates the HTTP response)
 - The thread writes the response to the connection
 - Finally, the thread become available to process another connection in the “todo” set

Disabling

`queue_requests`



The `queue_requests` option is `true` by default, enabling the separate thread used to buffer requests as described above.

If set to `false`, this buffer will not be used for connections while waiting for the request to arrive.

In this mode, when a connection is accepted, it is added to the “todo” queue immediately, and a worker will synchronously do any waiting necessary to read the HTTP request from the socket.

Deployment engineering for puma

Deployment engineering for puma

Puma is software that is expected to be run in a deployed environment eventually. You can certainly use it as your dev server only, but most people look to use it in their production deployments as well.

To that end, this is meant to serve as a foundation of wisdom how to do that in a way that increases happiness and decreases downtime.

Specifying puma

Most people want to do this by putting `gem "puma"` into their Gemfile, so we'll go ahead and assume that. Go add it now... we'll wait.

Welcome back!

Single vs Cluster mode

Puma was originally conceived as a thread-only webserver, but grew the ability to also use processes in version 2.

Here are some rules of thumb:

MRI

- Use cluster mode and set the number of workers to 1.5x the number of cpu cores in the machine, minimum 2.
- Set the number of threads to desired concurrent requests / number of workers. Puma defaults to 16 and that's a decent number.

Migrating from Unicorn

- If you're migrating from unicorn though, here are some settings to start with:
 - Set workers to half the number of unicorn workers you're using
 - Set threads to 2
 - Enjoy 50% memory savings
- As you grow more confident in the thread safety of your app, you can tune the workers down and the threads up.

Worker utilization

How do you know if you're got enough (or too many workers)?

A good question. Due to MRI's GIL, only one thread can be executing Ruby code at a time.

But since so many apps are waiting on IO from DBs, etc., they can utilize threads to make better use of the process.

The rule of thumb is you never want processes that are pegged all the time. This means that there is more work to do than the process can get through. On the other hand, if you have processes that sit around doing nothing, then they're just eating up resources.

Watching your CPU utilization over time and aim for about 70% on average. This means you've got capacity still but aren't starving threads.

Daemonizing

I prefer to not daemonize my servers and use something like `runit` or `upstart` to monitor them as child processes. This gives them fast response to crashes and makes it easy to figure out what is going on. Additionally, unlike `unicorn`, puma does not require daemonization to do zero-downtime restarts.

I see people using daemonization because they start puma directly via capistrano task and thus want it to live on past the `cap deploy`. To this people I said: You need to be using a process monitor. Nothing is making sure puma stays up in this scenario! You're just waiting for something weird to happen, puma to die, and to get paged at 3am. Do yourself a favor, at least the process monitoring your OS comes with, be it `sysvinit`, `upstart`, or `systemd`. Or branch out and use `runit` or hell, even `monit`.

Restarting

You probably will want to deploy some new code at some point, and you'd like puma to start running that new code. Minimizing the amount of time the server is unavailable would be nice as well. Here's how to do it:

1. Don't use `preload!`. This dirties the master process and means it will have to shutdown all the workers and re-exec itself to get your new code. It is not compatible with phased-restart and `prune_bundler` as well.
2. Use `prune_bundler`. This makes it so that the cluster master will detach itself from a Bundler context on start. This allows the cluster workers to load your app and start a brand new Bundler context within the worker only. This means your master remains pristine and can live on between new releases of your code.
3. Use phased-restart (`SIGUSR1` or `pumactl phased-restart`). This tells the master to kill off one worker at a time and restart them in your new code. This minimizes downtime and staggers the restart nicely. **WARNING** This means that both your

old code and your new code will be running concurrently. Most deployment solutions already cause that, but it's worth warning you about it again. Be careful with your migrations, etc!

Nginx configuration example file

Nginx configuration example file

This is a very common setup using an upstream. It was adapted from some Capistrano recipe I found on the Internet a while ago.

```
1. upstream myapp {
2.     server unix:///myapp/tmp/puma.sock;
3. }
4.
5. server {
6.     listen 80;
7.     server_name myapp.com;
8.
9.     # ~2 seconds is often enough for most folks to parse HTML/CSS and
10.    # retrieve needed images/icons/frames, connections are cheap in
11.    # nginx so increasing this is generally safe...
12.    keepalive_timeout 5;
13.
14.    # path for static files
15.    root /myapp/public;
16.    access_log /myapp/log/nginx.access.log;
17.    error_log /myapp/log/nginx.error.log info;
18.
19.    # this rewrites all the requests to the maintenance.html
20.    # page if it exists in the doc root. This is for capistrano's
21.    # disable web task
22.    if (-f $document_root/maintenance.html) {
23.        rewrite ^(.*)$ /maintenance.html last;
24.        break;
25.    }
26.
27.    location / {
28.        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
29.        proxy_set_header Host $http_host;
30.
31.        # If the file exists as a static file serve it directly without
32.        # running all the other rewrite tests on it
33.        if (-f $request_filename) {
34.            break;
35.        }
36.
37.        # check for index.html for directory index
38.        # if it's there on the filesystem then rewrite
39.        # the url to add /index.html to the end of it
40.        # and then break to send it to the next config rules.
41.        if (-f $request_filename/index.html) {
42.            rewrite (.*?) $1/index.html break;
```

```
43.     }
44.
45.     # this is the meat of the rack page caching config
46.     # it adds .html to the end of the url and then checks
47.     # the filesystem for that file. If it exists, then we
48.     # rewrite the url to have explicit .html on the end
49.     # and then send it on its way to the next config rule.
50.     # if there is no file on the fs then it sets all the
51.     # necessary headers and proxies to our upstream pumas
52.     if (-f $request_filename.html) {
53.         rewrite (.* ) $1.html break;
54.     }
55.
56.     if (!-f $request_filename) {
57.         proxy_pass http://myapp;
58.         break;
59.     }
60. }
61.
62. # Now this supposedly should work as it gets the filenames with querystrings that Rails provides.
63. # BUT there's a chance it could break the ajax calls.
64. location ~* \.(ico|css|gif|jpe?g|png|js)(\?[0-9]+)?$ {
65.     expires max;
66.     break;
67. }
68.
69. # Error pages
70. # error_page 500 502 503 504 /500.html;
71. location = /500.html {
72.     root /myapp/current/public;
73. }
74. }
```

Plugins

Plugins

Puma 3.0 added support for plugins that can augment configuration and service operations.

2 canonical plugins to look to aid in development of further plugins:

- `tmp_restart`: Restarts the server if the file `tmp/restart.txt` is touched
- `heroku`: Packages up the default configuration used by puma on Heroku

Plugins are activated in a puma configuration file (such as `config/puma.rb`) by adding `plugin "name"`, such as `plugin "heroku"`.

Plugins are activated based simply on path requirements so, activating the `heroku` plugin will simply be doing `require "puma/plugin/heroku"`. This allows gems to provide multiple plugins (as well as unrelated gems to provide puma plugins).

The `tmp_restart` plugin is bundled with puma, so it can always be used.

To use the `heroku` plugin, add `puma-heroku` to your Gemfile or install it.

API

At present, there are 2 hooks that plugins can use: `start` and `config`.

`start` runs when the server has started and allows the plugin to start other functionality to augment puma.

`config` runs when the server is being configured and is passed a `Puma::DSL` object that can be used to add additional configuration.

Any public methods in `Puma::Plugin` are the public API that any plugin may use.

In the future, more hooks and APIs will be added.

Restarts

Restarts

To perform a restart, there are 3 builtin mechanisms:

- Send the `puma` process the `SIGUSR2` signal
- Send the `puma` process the `SIGUSR1` signal (rolling restart, cluster mode only)
- Use the status server and issue `/restart`

No code is shared between the current and restarted process, so it should be safe to issue a restart any place where you would manually stop Puma and start it again.

If the new process is unable to load, it will simply exit. You should therefore run Puma under a process monitor (see below) when using it in production.

Normal vs Hot vs Phased Restart

A hot restart means that no requests will be lost while deploying your new code, since the server socket is kept open between restarts.

But beware, hot restart does not mean that the incoming requests won't hang for multiple seconds while your new code has not fully deployed. If you need a zero downtime and zero hanging requests deploy, you must use phased restart.

When you run `pumactl phased-restart`, Puma kills workers one-by-one, meaning that at least another worker is still available to serve requests, which lead to zero hanging requests (yay!).

But again beware, upgrading an application sometimes involves upgrading the database schema. With phased restart, there may be a moment during the deployment where processes belonging to the previous version and processes belonging to the new version both exist at the same time. Any database schema upgrades you perform must therefore be backwards-compatible with the old application version.

If you perform a lot of database migrations, you probably should not use phased restart and use a normal/hot restart instead (`pumactl restart`). That way, no code is shared while deploying (in that case, `preload_app!` might help for quicker deployment, see [“Clustered Mode” in the README](#)).

Release Directory

If your symlink releases into a common working directory (i.e., `/current` from Capistrano), Puma won't pick up your new changes when running phased restarts without additional configuration. You should set your working directory within Puma's config

to specify the directory it should use. This is a change from earlier versions of Puma (< 2.15) that would infer the directory for you.

```
1. # config/puma.rb
2.
3. directory '/var/www/current'
```

Cleanup Code

Puma isn't able to understand all the resources that your app may use, so it provides a hook in the configuration file you pass to `-C` called `on_restart`. The block passed to `on_restart` will be called, unsurprisingly, just before Puma restarts itself.

You should place code to close global log files, redis connections, etc. in this block so that their file descriptors don't leak into the restarted process. Failure to do so will result in slowly running out of descriptors and eventually obscure crashes as the server is restarted many times.

Sending Signals

The `unix signal` is a method of sending messages between `processes`). When a signal is sent, the operating system interrupts the target process's normal flow of execution. There are standard signals that are used to stop a process but there are also custom signals that can be used for other purposes. This document is an attempt to list all supported signals that Puma will respond to. In general, signals need only be sent to the master process of a cluster.

Sending Signals

If you are new to signals it can be useful to see how they can be used. When a process is created in a *nix like operating system it will have a `PID - or process identifier` that can be used to send signals to the process. For demonstration we will create an infinitely running process by tailing a file:

```
1. $ echo "foo" >> my.log
2. $ irb
3. > pid = Process.spawn 'tail -f my.log'
```

From here we can see that the tail process is running by using the `ps` command:

```
1. $ ps aux | grep tail
2. schneems      87152   0.0  0.0  2432772   492 s032  S+   12:46PM   0:00.00 tail -f my.log
```

You can send a signal in Ruby using the `Process module`:

```
1. $ irb
2. > puts pid
3. => 87152
4. Process.detach(pid) # http://ruby-doc.org/core-2.1.1/Process.html#method-c-detach
5. Process.kill("TERM", pid)
```

Now you will see via `ps` that there is no more `tail` process. Sometimes when referring to signals the `SIG` prefix will be used for instance `SIGTERM` is equivalent to sending `TERM` via `Process.kill`.

Puma Signals

Puma cluster responds to these signals:

- `TTIN` increment the worker count by 1
- `TTOU` decrement the worker count by 1
- `TERM` send `TERM` to worker. Worker will attempt to finish then exit.
- `USR2` restart workers. This also reloads puma configuration file, if there is one.

- `USR1` restart workers in phases, a rolling restart. This will not reload configuration file.
- `HUP` reopen log files defined in `stdout_redirect` configuration parameter. If there is no `stdout_redirect` option provided it will behave like `INT`
- `INT` equivalent of sending Ctrl-C to cluster. Will attempt to finish then exit.
- `CHLD`

Callbacks order in case of different signals

Start application

```

1. puma configuration file reloaded, if there is one
2. * Pruning Bundler environment
3. puma configuration file reloaded, if there is one
4.
5. before_fork
6. on_worker_fork
7. after_worker_fork
8.
9. Gemfile in context
10.
11. on_worker_boot
12.
13. Code of the app is loaded and running

```

Send USR2

```

1. on_worker_shutdown
2. on_restart
3.
4. puma configuration file reloaded, if there is one
5.
6. before_fork
7. on_worker_fork
8. after_worker_fork
9.
10. Gemfile in context
11.
12. on_worker_boot
13.
14. Code of the app is loaded and running

```

Send USR1

```

1. on_worker_shutdown
2. on_worker_fork
3. after_worker_fork

```


- 4.
5. `Gemfile` in context
- 6.
7. `on_worker_boot`
- 8.
9. `Code` of the app `is` loaded `and` running

Service Configuration

systemd

`systemd` is a commonly available init system (PID 1) on many Linux distributions. It offers process monitoring (including automatic restarts) and other useful features for running Puma in production.

Service Configuration

Below is a sample `puma.service` configuration file for `systemd`, which can be copied or symlinked to `/etc/systemd/system/puma.service`, or if desired, using an application or instance specific name.

Note that this uses the `systemd` preferred “simple” type where the start command remains running in the foreground (does not fork and exit). See also, the [Alternative Forking Configuration](#) below.

```

1. [Unit]
2. Description=Puma HTTP Server
3. After=network.target
4.
5. # Uncomment for socket activation (see below)
6. # Requires=puma.socket
7.
8. [Service]
9. # Foreground process (do not use --daemon in ExecStart or config.rb)
10. Type=simple
11.
12. # Preferably configure a non-privileged user
13. # User=
14.
15. # The path to the puma application root
16. # Also replace the "<WD>" place holders below with this path.
17. WorkingDirectory=
18.
19. # Helpful for debugging socket activation, etc.
20. # Environment=PUMA_DEBUG=1
21.
22. # The command to start Puma. This variant uses a binstub generated via
23. # `bundle binstubs puma --path ./sbin` in the WorkingDirectory
24. # (replace "<WD>" below)
25. ExecStart=<WD>/sbin/puma -b tcp://0.0.0.0:9292 -b ssl://0.0.0.0:9293?key=key.pem&cert=cert.pem
26.
```

```

27. # Variant: Use config file with `bind` directives instead:
28. # ExecStart=<WD>/sbin/puma -C config.rb
29. # Variant: Use `bundle exec --keep-file-descriptors puma` instead of binstub
30.
31. Restart=always
32.
33. [Install]
34. WantedBy=multi-user.target

```

See [systemd.exec](#) for additional details.

Socket Activation

systemd and puma also support socket activation, where systemd opens the listening socket(s) in advance and provides them to the puma master process on startup. Among other advantages, this keeps listening sockets open across puma restarts and achieves graceful restarts, including when upgraded puma, and is compatible with both clustered mode and application preload.

Note: Socket activation doesn't currently work on jruby. This is tracked in [#1367](#).

To use socket activation, configure one or more `ListenStream` sockets in a companion `*.socket` unit file. Also uncomment the associated `Requires` directive for the socket unit in the service file (see above.) Here is a sample puma.socket, matching the ports used in the above puma.service:

```

1. [Unit]
2. Description=Puma HTTP Server Accept Sockets
3.
4. [Socket]
5. ListenStream=0.0.0.0:9292
6. ListenStream=0.0.0.0:9293
7.
8. # AF_UNIX domain socket
9. # SocketUser, SocketGroup, etc. may be needed for Unix domain sockets
10. # ListenStream=/run/puma.sock
11.
12. # Socket options matching Puma defaults
13. NoDelay=true
14. ReusePort=true
15. Backlog=1024
16.
17. [Install]
18. WantedBy=sockets.target

```

See [systemd.socket](#)

for additional configuration details.

Note that the above configurations will work with Puma in either single process or cluster mode.

Sockets and symlinks

When using releases folders, you should set the socket path using the shared folder path (ex. `/srv/projet/shared/tmp/puma.sock`), not the release folder path (`/srv/projet/releases/1234/tmp/puma.sock`).

Puma will detect the release path socket as different than the one provided by systemd and attempt to bind it again, resulting in the exception

```
There is already a server bound to: .
```

Usage

Without socket activation, use `systemctl` as root (e.g. via `sudo`) as with other system services:

```
1. # After installing or making changes to puma.service
2. systemctl daemon-reload
3.
4. # Enable so it starts on boot
5. systemctl enable puma.service
6.
7. # Initial start up.
8. systemctl start puma.service
9.
10. # Check status
11. systemctl status puma.service
12.
13. # A normal restart. Warning: listeners sockets will be closed
14. # while a new puma process initializes.
15. systemctl restart puma.service
```

With socket activation, several but not all of these commands should be run for both socket and service:

```
1. # After installing or making changes to either puma.socket or
2. # puma.service.
3. systemctl daemon-reload
4.
5. # Enable both socket and service so they start on boot. Alternatively
6. # you could leave puma.service disabled and systemd will start it on
7. # first use (with startup lag on first request)
8. systemctl enable puma.socket puma.service
9.
```

```

10. # Initial start up. The Requires directive (see above) ensures the
11. # socket is started before the service.
12. systemctl start puma.socket puma.service
13.
14. # Check status of both socket and service.
15. systemctl status puma.socket puma.service
16.
17. # A "hot" restart, with systemd keeping puma.socket listening and
18. # providing to the new puma (master) instance.
19. systemctl restart puma.service
20.
21. # A normal restart, needed to handle changes to
22. # puma.socket, such as changing the ListenStream ports. Note
23. # daemon-reload (above) should be run first.
24. systemctl restart puma.socket puma.service

```

Here is sample output from `systemctl status` with both service and socket running:

```

1. ● puma.socket - Puma HTTP Server Accept Sockets
2.   Loaded: loaded (/etc/systemd/system/puma.socket; enabled; vendor preset: enabled)
3.   Active: active (running) since Thu 2016-04-07 08:40:19 PDT; 1h 2min ago
4.   Listen: 0.0.0.0:9233 (Stream)
5.           0.0.0.0:9234 (Stream)
6.
7. Apr 07 08:40:19 hx systemd[874]: Listening on Puma HTTP Server Accept Sockets.
8.
9. ● puma.service - Puma HTTP Server
10.   Loaded: loaded (/etc/systemd/system/puma.service; enabled; vendor preset: enabled)
11.   Active: active (running) since Thu 2016-04-07 08:40:19 PDT; 1h 2min ago
12.   Main PID: 28320 (ruby)
13.   CGroup: /system.slice/puma.service
14.           └─28320 puma 3.3.0 (tcp://0.0.0.0:9233,ssl://0.0.0.0:9234?key=key.pem&cert=cert.pem) [app]
15.           └─28323 puma: cluster worker 0: 28320 [app]
16.           └─28327 puma: cluster worker 1: 28320 [app]
17.
18. Apr 07 08:40:19 hx puma[28320]: Puma starting in cluster mode...
19. Apr 07 08:40:19 hx puma[28320]: * Version 3.3.0 (ruby 2.2.4-p230), codename: Jovial Platypus
20. Apr 07 08:40:19 hx puma[28320]: * Min threads: 0, max threads: 16
21. Apr 07 08:40:19 hx puma[28320]: * Environment: production
22. Apr 07 08:40:19 hx puma[28320]: * Process workers: 2
23. Apr 07 08:40:19 hx puma[28320]: * Phased restart available
24. Apr 07 08:40:19 hx puma[28320]: * Activated tcp://0.0.0.0:9233
25. Apr 07 08:40:19 hx puma[28320]: * Activated ssl://0.0.0.0:9234?key=key.pem&cert=cert.pem
26. Apr 07 08:40:19 hx puma[28320]: Use Ctrl-C to stop

```

Alternative Forking Configuration

Other systems/tools might expect or need puma to be run as a "traditional" forking server, for example so that the `pumactl`

command can be used directly and outside of systemd for stop/start/restart. This use case is incompatible with systemd socket activation, so it should not be configured. Below is an alternative puma.service config sample, using `Type=forking` and the `--daemon` flag in `ExecStart`. Here systemd is playing a role more equivalent to SysV init.d, where it is responsible for starting Puma on boot (multi-user.target) and stopping it on shutdown, but is not performing continuous restarts. Therefore running Puma in cluster mode, where the master can restart workers, is highly recommended. See the systemd [Restart](#) directive for details.

```

1. [Unit]
2. Description=Puma HTTP Forking Server
3. After=network.target
4.
5. [Service]
6. # Background process configuration (use with --daemon in ExecStart)
7. Type=forking
8.
9. # Preferably configure a non-privileged user
10. # User=
11.
12. # The path to the puma application root
13. # Also replace the "<WD>" place holders below with this path.
14. WorkingDirectory=
15.
16. # The command to start Puma
17. # (replace "<WD>" below)
18. ExecStart=bundle exec puma -C <WD>/shared/puma.rb --daemon
19.
20. # The command to stop Puma
21. # (replace "<WD>" below)
22. ExecStop=bundle exec pumactl -S <WD>/shared/tmp/pids/puma.state stop
23.
24. # Path to PID file so that systemd knows which is the master process
25. PIDFile=<WD>/shared/tmp/pids/puma.pid
26.
27. # Should systemd restart puma?
28. # Use "no" (the default) to ensure no interference when using
29. # stop/start/restart via `pumactl`. The "on-failure" setting might
30. # work better for this purpose, but you must test it.
31. # Use "always" if only `systemctl` is used for start/stop/restart, and
32. # reconsider if you actually need the forking config.
33. Restart=no
34.
35. [Install]
36. WantedBy=multi-user.target

```

capistrano3-puma

By default,

`capistrano3-puma` uses

`pumactl` for deployment restarts, outside of `systemd`. To learn the exact commands that this tool would use for `ExecStart` and `ExecStop`, use the following `cap` commands in dry-run mode, and update from the above forking service configuration accordingly. Note also that the configured `User` should likely be the same as the `capistrano3-puma` `:puma_user` option.

1. `stage=production # or different stage, as needed`
2. `cap $stage puma:start --dry-run`
3. `cap $stage puma:stop --dry-run`

Tools

- [Puma daemon service](#)
- [Puma as a service using rc.d](#)
- [Puma as a service](#)
- [Puma as a service using Upstart](#)

Puma daemon service

Puma daemon service

Deprecatation Warning : `init.d` was replaced by `systemd` since Debian 8 and Ubuntu 16.04, you should look into </docs/systemd> unless you are on an older OS.

Init script to manage multiple Puma servers on the same box using start-stop-daemon.

Installation

```
1. # Copy the init script to services directory
2. sudo cp puma /etc/init.d
3. sudo chmod +x /etc/init.d/puma
4.
5. # Make it start at boot time.
6. sudo update-rc.d -f puma defaults
7.
8. # Copy the Puma runner to an accessible location
9. sudo cp run-puma /usr/local/bin
10. sudo chmod +x /usr/local/bin/run-puma
11.
12. # Create an empty configuration file
13. sudo touch /etc/puma.conf
```

Managing the jungle

Puma apps are held in `/etc/puma.conf` by default. It's mainly a CSV file and every line represents one app. Here's the syntax:

```
1. app-path,user,config-file-path,log-file-path,environment-variables
```

You can add an instance by editing the file or running the following command:

```
1. sudo /etc/init.d/puma add /path/to/app user /path/to/app/config/puma.rb /path/to/app/log/puma.log
```

The config and log paths, as well as the environment variables, are optional parameters and default to:

- config: `/path/to/app/config/puma.rb`
- log: `/path/to/app/log/puma.log`
- environment: (empty)

Multiple environment variables need to be separated by a semicolon, e.g.

```
1. FOO=1;BAR=2
```

To remove an app, simply delete the line from the config file or run:

```
1. sudo /etc/init.d/puma remove /path/to/app
```

The command will make sure the Puma instance stops before removing it from the jungle.

Assumptions

- The script expects a temporary folder named `/path/to/app/tmp/puma` to exist. Create it if it's not there by default. The pid and state files should live there and must be called: `tmp/puma/pid` and `tmp/puma/state`. You can change those if you want but you'll have to adapt the script for it to work.
- Here's what a minimal app's config file should have:

```
1. pidfile "/path/to/app/tmp/puma/pid"
2. state_path "/path/to/app/tmp/puma/state"
3. activate_control_app
```

Puma as a service using rc.d

Puma as a service using rc.d

Manage multiple Puma servers as services on one box using FreeBSD's rc.d service.

Dependencies

- `jq` - a command-line json parser is needed to parse the json in the config file

Installation

```
1. # Copy the puma script to the rc.d directory (make sure everyone has read/execute perms)
2. sudo cp puma /usr/local/etc/rc.d/
3.
4. # Create an empty configuration file
5. sudo touch /usr/local/etc/puma.conf
6.
7. # Enable the puma service
8. sudo echo 'puma_enable="YES"' >> /etc/rc.conf
```

Managing the jungle

Puma apps are referenced in `/usr/local/etc/puma.conf` by default.

Start the jungle running:

```
service puma start
```

This script will run at boot time.

You can also stop the jungle (stops ALL puma instances) by running:

```
service puma stop
```

To restart the jungle:

```
service puma restart
```

Conventions

- The script expects:
 - a config file to exist under `config/puma.rb` in your app. E.g.: `/home/apps/my-app/config/puma.rb`.

You can always change those defaults by editing the scripts.

Here's what a minimal app's config file should have

```
1. {  
2.   "servers" : [  
3.     {  
4.       "dir": "/path/to/rails/project",  
5.       "user": "deploy-user",  
6.       "ruby_version": "ruby.version",  
7.       "ruby_env": "rbenv"  
8.     }  
9.   ]  
10. }
```

Before starting...

You need to customise `puma.conf` to:

- Set the right user your app should be running on unless you want root to execute it!
- Set the directory of the app
- Set the ruby version to execute
- Set the ruby environment (currently set to rbenv, since that is the only ruby environment currently supported)
- Add additional server instances following the scheme in the example

Notes:

Only rbenv is currently supported.

Puma as a service

Puma as a service

Upstart

See `/tools/jungle/upstart` for Ubuntu's upstart scripts.

Systemd

See </docs/systemd>.

Init.d

Deprecation Warning : `init.d` was replaced by `systemd` since Debian 8 and Ubuntu 16.04, you should look into </docs/systemd> unless you are on an older OS.

See `/tools/jungle/init.d` for tools to use with init.d and start-stop-daemon.

rc.d

See `/tools/jungle/rc.d` for FreeBSD's rc.d scripts

Puma as a service using Upstart

Puma as a service using Upstart

Manage multiple Puma servers as services on the same box using Ubuntu upstart.

Installation

```
1. # Copy the scripts to services directory
2. sudo cp puma.conf puma-manager.conf /etc/init
3.
4. # Create an empty configuration file
5. sudo touch /etc/puma.conf
```

Managing the jungle

Puma apps are referenced in `/etc/puma.conf` by default. Add each app's path as a new line, e.g.:

```
1. /home/apps/my-cool-ruby-app
2. /home/apps/another-app/current
```

Start the jungle running:

```
sudo start puma-manager
```

This script will run at boot time.

Start a single puma like this:

```
sudo start puma app=/path/to/app
```

Logs

Everything is logged by upstart, defaulting to `/var/log/upstart`.

Each puma instance is named after its directory, so for an app called `/home/apps/my-app` the log file would be `/var/log/upstart/puma-_home_apps_my-app.log`.

Conventions

- The script expects:
 - a config file to exist under `config/puma.rb` in your app. E.g.: `/home/apps/my-`

```
app/config/puma.rb
```

 .

- a temporary folder to put the PID, socket and state files to exist called `tmp/puma` . E.g.: `/home/apps/my-app/tmp/puma` . Puma will take care of the files for you.

You can always change those defaults by editing the scripts.

Here's what a minimal app's config file should have

```
1. pidfile "/path/to/app/tmp/puma/pid"
2. state_path "/path/to/app/tmp/puma/state"
3. activate_control_app
```

Before starting...

You need to customise `puma.conf` to:

- Set the right user your app should be running on unless you want root to execute it!
 - Look for `setuid apps` and `setgid apps` , uncomment those lines and replace `apps` to whatever your deployment user is.
 - Replace `apps` on the paths (or set the right paths to your user's home) everywhere else.
- Uncomment the source lines for `rbenv` or `rvm` support unless you use a system wide installation of Ruby.

HISTORY(升级日志)

3.12.0 / 2018-07-13

- 5 features:
 - You can now specify which SSL ciphers the server should support, default is unchanged (#1478)
 - The setting for Puma's `max_threads` is now in `Puma.stats` (#1604)
 - Pool capacity is now in `Puma.stats` (#1579)
 - Installs restricted to Ruby 2.2+ (#1506)
 - `--control` is now deprecated in favor of `--control-url` (#1487)
- 2 bugfixes:
 - Workers will no longer accept more web requests than they have capacity to process. This prevents an issue where one worker would accept lots of requests while starving other workers (#1563)
 - In a test env puma now emits the stack on an exception (#1557)

3.11.4 / 2018-04-12

- 2 features:
 - Manage puma as a service using rc.d (#1529)
 - Server stats are now available from a top level method (#1532)
- 5 bugfixes:
 - Fix parsing CLI options (#1482)
 - Order of stderr and stdout is made before redirecting to a log file (#1511)
 - Init.d fix of `ps -p` to check if pid exists (#1545)
 - Early hints bugfix (#1550)
 - Purge interrupt queue when closing socket fails (#1553)

3.11.3 / 2018-03-05

- 3 bugfixes:
 - Add closed? to MiniSSL::Socket for use in reactor (#1510)
 - Handle EOFError at the toplevel of the server threads (#1524) (#1507)
 - Deal with zero sized bodies when using SSL (#1483)

3.11.2 / 2018-01-19

- 1 bugfix:
 - Deal with read_nonblock returning nil early

3.11.1 / 2018-01-18

- 1 bugfix:
 - Handle read_nonblock returning nil when the socket close (#1502)

3.11.0 / 2017-11-20

- 2 features:
 - HTTP 103 Early Hints (#1403)
 - 421/451 status codes now have correct status messages attached (#1435)
- 9 bugfixes:
 - Environment config files (/config/puma/.rb) load correctly (#1340)
 - Specify windows dependencies correctly (#1434, #1436)
 - puma/events required in test helper (#1418)
 - Correct control CLI's option help text (#1416)
 - Remove a warning for unused variable in mini_ssl (#1409)
 - Correct pumactl docs argument ordering (#1427)
 - Fix an uninitialized variable warning in server.rb (#1430)
 - Fix docs typo/error in Launcher init (#1429)
 - Deal with leading spaces in RUBYOPT (#1455)
- 2 other:
 - Add docs about internals (#1425, #1452)
 - Tons of test fixes from [@MSP-Greg](#) (#1439, #1442, #1464)

3.10.0 / 2017-08-17

- 3 features:
 - The status server has a new /gc and /gc-status command. (#1384)
 - The persistent and first data timeouts are now configurable (#1111)
 - Implemented RFC 2324 (#1392)
- 12 bugfixes:
 - Not really a Puma bug, but [@NickolasVashchenko](#) created a gem to workaround a Ruby bug that some users of Puma may be experiencing. See README for more. (#1347)
 - Fix hangups with SSL and persistent connections. (#1334)
 - Fix Rails double-binding to a port (#1383)
 - Fix incorrect thread names (#1368)
 - Fix issues with /etc/hosts and JRuby where localhost addresses were not correct. (#1318)

- Fix compatibility with RUBYOPT="--enable-frozen-string-literal" (#1376)
- Fixed some compiler warnings (#1388)
- We actually run the integration tests in CI now (#1390)
- No longer shipping unnecessary directories in the gemfile (#1391)
- If RUBYOPT is nil, we no longer blow up on restart. (#1385)
- Correct response to SIGINT (#1377)
- Proper exit code returned when we receive a TERM signal (#1337)
- 3 refactors:
 - Various test improvements from @grosser
 - Rubocop (#1325)
 - Hoe has been removed (#1395)
- 1 known issue:
 - Socket activation doesn't work in JRuby. Their fault, not ours. (#1367)

3.9.1 / 2017-06-03

- 2 bugfixes:
 - Fixed compatibility with older Bundler versions (#1314)
 - Some internal test/development cleanup (#1311, #1313)

3.9.0 / 2017-06-01

- 2 features:
 - The ENV is now reset to its original values when Puma restarts via USR1/USR2 (#1260) (MRI only, no JRuby support)
 - Puma will no longer accept more clients than the maximum number of threads. (#1278)
- 9 bugfixes:
 - Reduce information leakage by preventing HTTP parse errors from writing environment hashes to STDERR (#1306)
 - Fix SSL/WebSocket compatibility (#1274)
 - HTTP headers with empty values are no longer omitted from responses. (#1261)
 - Fix a Rack env key which was set to nil. (#1259)
 - peercert has been implemented for JRuby (#1248)
 - Fix port settings when using rails s (#1277, #1290)
 - Fix compat w/LibreSSL (#1285)
 - Fix restarting Puma w/symlinks and a new Gemfile (#1282)
 - Replace Dir.exists? with Dir.exist? (#1294)

- 1 known issue:

- A bug in MRI 2.2+ can result in `IOError: stream closed`. See #1206. This issue has existed since at least Puma 3.6, and probably further back.

- 1 refactor:

- Lots of test fixups from [@grosser](#).

3.8.2 / 2017-03-14

- 1 bugfix:

- Deal with `getsockopt` with `TCP_INFO` failing for sockets that say they're TCP but aren't really. (#1241)

3.8.1 / 2017-03-10

- 1 bugfix:

- Remove method call to method that no longer exists (#1239)

3.8.0 / 2017-03-09

- 2 bugfixes:

- Port from rack handler does not take precedence over config file in Rails 5.1.0.beta2+ and 5.0.1.rc3+ (#1234)
- The `tmp/restart.txt` plugin no longer restricts the user from running more than one server from the same folder at a time (#1226)

- 1 feature:

- Closed clients are aborted to save capacity (#1227)

- 1 refactor:

- Bundler is no longer a dependency from tests (#1213)

3.7.1 / 2017-02-20

- 2 bugfixes:

- Fix typo which blew up MiniSSL (#1182)
- Stop overriding command-line options with the config file (#1203)

3.7.0 / 2017-01-04

- 6 minor features:

- Allow rack handler to accept ssl host. (#1129)

- Refactor TIOU processing. TIOU now handles multiple signals at once. (#1165)
- Pickup any remaining chunk data as the next request.
- Prevent short term thread churn - increased auto trim default to 30 seconds.
- Raise error when `stdout` or `stderr` is not writable. (#1175)
- Add Rack 2.0 support to gemspec. (#1068)

- 5 refactors:

- Compare host and server name only once per call. (#1091)
- Minor refactor on Thread pool (#1088)
- Removed a ton of unused constants, variables and files.
- Use MRI macros when allocating heap memory
- Use hooks for on_booted event. (#1160)

- 14 bugfixes:

- Add eof? method to NullIO? (#1169)
- Fix Puma startup in provided init.d script (#1061)
- Fix default SSL mode back to none. (#1036)
- Fixed the issue of `@listeners` getting nil io (#1120)
- Make `get_dh1024` compatible with OpenSSL v1.1.0 (#1178)
- More gracefully deal with SSL sessions. Fixes #1002
- Move puma.rb to just autoloader. Fixes #1063
- MiniSSL: Provide write as <<. Fixes #1089
- Prune bundler should inherit fds (#1114)
- Replace use of Process.getpgid which does not behave as intended on all platforms (#1110)
- Transfer encoding header should be downcased before comparison (#1135)
- Use same write log logic for hijacked requests. (#1081)
- Fix `uninitialized constant Puma::StateFile` (#1138)
- Fix access priorities of each level in LeveledOptions (#1118)

- 3 others:

- Lots of tests added/fixed/improved. Switched to Minitest from Test::Unit. Big thanks to [@frodsan](#).
- Lots of documentation added/improved.
- Add license indicators to the HTTP extension. (#1075)

3.6.2 / 2016-11-22

- 1 bug fix:

- Revert #1118/Fix access priorities of each level in LeveledOptions. This had an unintentional side effect of changing the importance of command line options, such as -p.

3.6.1 / 2016-11-21

- 8 bug fixes:
 - Fix Puma start in init.d script.
 - Fix default SSL mode back to none. Fixes #1036
 - Fixed the issue of [@listeners](#) getting nil io, fix rails restart (#1120)
 - More gracefully deal with SSL sessions. Fixes #1002
 - Prevent short term thread churn.
 - Provide write as <<. Fixes #1089
 - Fix access priorities of each level in LeveledOptions - fixes TTIN.
 - Stub description files updated for init.d.
- 2 new project committers:
 - Nate Berkopec ([@nateberkopec](#))
 - Richard Schneeman ([@schneems](#))

3.6.0 / 2016-07-24

- 12 bug fixes:
 - Add ability to detect a shutting down server. Fixes #932
 - Add support for Expect: 100-continue. Fixes #519
 - Check SSLContext better. Fixes #828
 - Clarify behavior of '-t num'. Fixes #984
 - Don't default to VERIFY_PEER. Fixes #1028
 - Don't use ENV['PWD'] on windows. Fixes #1023
 - Enlarge the scope of catching app exceptions. Fixes #1027
 - Execute background hooks after daemonizing. Fixes #925
 - Handle HUP as a stop unless there is IO redirection. Fixes #911
 - Implement chunked request handling. Fixes #620
 - Just rescue exception to return a 500. Fixes #1027
 - Redirect IO in the jruby daemon mode. Fixes #778

3.5.2 / 2016-07-20

- 1 bug fix:
 - Don't let persistent_timeout be nil
- 1 PR merged:
 - Merge pull request #1021 from benzrf/patch-1

3.5.1 / 2016-07-20

- 1 bug fix:
 - Be sure to only listen on host:port combos once. Fixes #1022

3.5.0 / 2016-07-18

- 1 minor features:
 - Allow `persistent_timeout` to be configured via the `dsl`.
- 9 bug fixes:
 - Allow a bare `%` in a query string. Fixes #958
 - Explicitly listen on all localhost addresses. Fixes #782
 - Fix `TCPLogger` log error in tcp cluster mode.
 - Fix puma/puma#968 Cannot bind SSL port due to missing `verify_mode` option
 - Fix puma/puma#968 Default `verify_mode` to `peer`
 - Log any exceptions in `ThreadPool`. Fixes #1010
 - Silence connection errors in the reactor. Fixes #959
 - Tiny fixes in hook documentation for #840
 - It should not log requests if we want it to be quiet
- 5 doc fixes:
 - Add How to stop Puma on Heroku using plugins to the example directory
 - Provide both hot and phased restart in jungle script
 - Update reference to the instances management script
 - Update default number of threads
 - Fix typo in example config
- 14 PRs merged:
 - Merge pull request #1007 from willnet/patch-1
 - Merge pull request #1014 from jeznnet/patch-1
 - Merge pull request #1015 from bf4/patch-1
 - Merge pull request #1017 from jorihardman/configurable_persistent_timeout
 - Merge pull request #954 from jf/master
 - Merge pull request #955 from jf/add-request-info-to-standard-error-rescue
 - Merge pull request #956 from maxkwallace/master
 - Merge pull request #960 from kmayer/kmayer-plugins-heroku-restart
 - Merge pull request #969 from frankwong15/master
 - Merge pull request #970 from willnet/delete-blank-document
 - Merge pull request #974 from rocketjob/feature/name_threads
 - Merge pull request #977 from snow/master
 - Merge pull request #981 from zach-chai/patch-1
 - Merge pull request #993 from scorix/master

3.4.0 / 2016-04-07

- 2 minor features:
 - Add ability to force threads to stop on shutdown. Fixes #938
 - Detect and commit seppuku when fork(2) fails. Fixes #529
- 3 unknowns:
 - Ignore errors trying to update the backport tables. Fixes #788
 - Invoke the `lowlevel_error` in more places to allow for exception tracking. Fixes #894
 - Update the query string when an absolute URI is used. Fixes #937
- 5 doc fixes:
 - Add Process Monitors section to top-level README
 - Better document the hooks. Fixes #840
 - docs/system.md sample config refinements and elaborations
 - Fix typos at couple of places.
 - Cleanup warnings
- 3 PRs merged:
 - Merge pull request #945 from dekellum/systemd-docs-refined
 - Merge pull request #946 from vipulnsward/rm-pid
 - Merge pull request #947 from vipulnsward/housekeeping-typos

3.3.0 / 2016-04-05

- 2 minor features:
 - Allow overriding options of Configuration object
 - Rename to `inherit_ssl_listener` like `inherit_tcp|unix`
- 2 doc fixes:
 - Add docs/systemd.md (with socket activation sub-section)
 - Document UNIX signals with cluster on README.md
- 3 PRs merged:
 - Merge pull request #936 from prathamesh-sonpatki/allow-overriding-config-options
 - Merge pull request #940 from kyledrake/signalsdoc
 - Merge pull request #942 from dekellum/socket-activate-improve

3.2.0 / 2016-03-20

- 1 deprecation removal:

- Delete capistrano.rb
- 3 bug fixes:
 - Detect gems.rb as well as Gemfile
 - Simplify and fix logic for directory to use when restarting for all phases
 - Speed up phased-restart start
- 2 PRs merged:
 - Merge pull request #927 from jlecour/gemfile_variants
 - Merge pull request #931 from joneslee85/patch-10

3.1.1 / 2016-03-17

- 4 bug fixes:
 - Disable USR1 usage on JRuby
 - Fixes #922 - Correctly define file encoding as UTF-8
 - Set a more explicit SERVER_SOFTWARE Rack variable
 - Show RUBY_ENGINE_VERSION if available. Fixes #923
- 3 PRs merged:
 - Merge pull request #912 from tricknotes/fix-allow-failures-in-travis-yml
 - Merge pull request #921 from swrobel/patch-1
 - Merge pull request #924 from tbrisker/patch-1

3.1.0 / 2016-03-05

- 1 minor feature:
 - Add 'import' directive to config file. Fixes #916
- 5 bug fixes:
 - Add 'fetch' to options. Fixes #913
 - Fix jruby daemonization. Fixes #918
 - Recreate the proper args manually. Fixes #910
 - Require 'time' to get iso8601. Fixes #914

3.0.2 / 2016-02-26

- 5 bug fixes:
 - Fix 'undefined local variable or method `pid` for #<_img src="http://www.emoji-cheat-sheet.com/graphics/emojis/ControlCLI.png" class="emoji" title=":ControlCLI:" alt=":ControlCLI:">0x007f185fcef968>' when

execute pumactl with `--pid` option.

- Fix 'undefined method `windows?` for Puma:Module' when execute pumactl.
- Harden tmp_restart against errors related to the restart file
- Make `plugin :tmp_restart` behavior correct in Windows.
- fix uninitialized constant Puma:::StateFile

- 3 PRs merged:

- Merge pull request #901 from mitto/fix-pumactl-uninitialized-constant-statefile
- Merge pull request #902 from corrupt952/fix_undefined_method_and_variable_when_execute_pumactl
- Merge pull request #905 from Eric-Guo/master

3.0.1 / 2016-02-25

- 1 bug fix:

- Removed the experimental support for `async.callback` as it broke websockets entirely. Seems no server has both `hijack` and `async.callback` and thus faye is totally confused what to do and doesn't work.

3.0.0 / 2016-02-25

- 2 major changes:

- Ruby pre-2.0 is no longer supported. We'll do our best to not add features that break those rubies but will no longer be testing with them.
- Don't log requests by default. Fixes #852

- 2 major features:

- Plugin support! Plugins can interact with configuration as well as provide augment server functionality!
- Experimental `env['async.callback']` support

- 4 minor features:

- Listen to unix socket with provided backlog if any
- Improves the clustered stats to report worker stats
- Pass the env to the `lowlevel_error` handler. Fixes #854
- Treat path-like hosts as unix sockets. Fixes #824

- 5 bug fixes:

- Clean thread locals when using `keepalive`. Fixes #823

- Cleanup compiler warnings. Fixes #815
- Expose closed? for use by the reactor. Fixes #835
- Move signal handlers to separate method to prevent space leak. Fixes #798
- Signal not full on worker exit #876
- 5 doc fixes:
 - Update README.md with various grammar fixes
 - Use newest version of Minitest
 - Add directory configuration docs, fix typo [ci skip]
 - Remove old COPYING notice. Fixes #849
- 10 merged PRs:
 - Merge pull request #871 from deepj/travis
 - Merge pull request #874 from wallclockbuilder/master
 - Merge pull request #883 from dadah89/igor/trim_only_worker
 - Merge pull request #884 from uistudio/async-callback
 - Merge pull request #888 from mlarraz/tick_minitest
 - Merge pull request #890 from todd/directory_docs
 - Merge pull request #891 from ctaintor/improve_clustered_status
 - Merge pull request #893 from spastorino/add_missing_require
 - Merge pull request #897 from zendesk/master
 - Merge pull request #899 from kch/kch-readme-fixes

2.16.0 / 2016-01-27

- 7 minor features:
 - Add 'set_remote_address' config option
 - Allow to run puma in silent mode
 - Expose cli options in DSL
 - Support passing JRuby keystore info in ssl_bind DSL
 - Allow umask for unix:///style control urls
 - Expose `old_worker_count` in stats url
 - Support TLS client auth (verify_mode) in jruby
- 7 bug fixes:
 - Don't persist before_fork hook in state file
 - Reload bundler before pulling in rack. Fixes #859
 - Remove NEWRELIC_DISPATCHER env variable
 - Cleanup C code
 - Use Timeout.timeout instead of Object.timeout
 - Make phased restarts faster
 - Ignore the case of certain headers, because HTTP
- 1 doc changes:

- Test against the latest Ruby 2.1, 2.2, 2.3, head and JRuby 9.0.4.0 on Travis
- 12 merged PRs
 - Merge pull request #822 from kwugirl/remove_NEWRELIC_DISPATCHER
 - Merge pull request #833 from joemiller/jruby-client-tls-auth
 - Merge pull request #837 from YuriSolovyov/ssl-keystore-jruby
 - Merge pull request #839 from mezuka/master
 - Merge pull request #845 from deepj/timeout-deprecation
 - Merge pull request #846 from sriedel/strip_before_fork
 - Merge pull request #850 from deepj/travis
 - Merge pull request #853 from Jeffrey6052/patch-1
 - Merge pull request #857 from zendesk/faster_phased_restarts
 - Merge pull request #858 from mlarraz/fix_some_warnings
 - Merge pull request #860 from zendesk/expose_old_worker_count
 - Merge pull request #861 from zendesk/allow_control_url_umask

2.15.3 / 2015-11-07

- 1 bug fix:
 - Fix JRuby parser

2.15.2 / 2015-11-06

- 2 bug fixes:
 - ext/puma_http11: handle duplicate headers as per RFC
 - Only set ctx.ca iff there is a params['ca'] to set with.
- 2 PRs merged:
 - Merge pull request #818 from unleashed/support-duplicate-headers
 - Merge pull request #819 from VictorLowther/fix-ca-and-verify_null-exception

2.15.1 / 2015-11-06

- 1 bug fix:
 - Allow older openssl versions

2.15.0 / 2015-11-06

- 6 minor features:
 - Allow setting ca without setting a verify mode

- Make jungle for init.d support rbenv
- Use SSL_CTX_use_certificate_chain_file for full chain
- cluster: add worker_boot_timeout option
- configuration: allow empty tags to mean no tag desired
- puma/cli: support specifying STD{OUT,ERR} redirections and append mode
- 5 bug fixes:
 - Disable SSL Compression
 - Fix bug setting worker_directory when using a symlink directory
 - Fix error message in DSL that was slightly inaccurate
 - Pumactl: set correct process name. Fixes #563
 - thread_pool: fix race condition when shutting down workers
- 10 doc fixes:
 - Add before_fork explanation in Readme.md
 - Correct spelling in DEPLOYMENT.md
 - Correct spelling in docs/nginx.md
 - Fix spelling errors.
 - Fix typo in deployment description
 - Fix typos (it's -> its) in events.rb and server.rb
 - fixing for typo mentioned in #803
 - Spelling correction for README
 - thread_pool: fix typos in comment
 - More explicit docs for worker_timeout
- 18 PRs merged:
 - Merge pull request #768 from nathansamson/patch-1
 - Merge pull request #773 from rossta/spelling_corrections
 - Merge pull request #774 from snow/master
 - Merge pull request #781 from sunsations/fix-typo
 - Merge pull request #791 from unleashed/allow_empty_tags
 - Merge pull request #793 from robdimarco/fix-working-directory-symlink-bug
 - Merge pull request #794 from peterkeen/patch-1
 - Merge pull request #795 from unleashed/redirects-from-cmdline
 - Merge pull request #796 from cschneid/fix_dsl_message
 - Merge pull request #799 from annafw/master
 - Merge pull request #800 from liamseanbrady/fix_typo
 - Merge pull request #801 from scottjg/ssl-chain-file
 - Merge pull request #802 from scottjg/ssl-crimes
 - Merge pull request #804 from burningTyger/patch-2
 - Merge pull request #809 from unleashed/threadpool-fix-race-in-shutdown
 - Merge pull request #810 from vlmonk/fix-pumactl-restart-bug
 - Merge pull request #814 from schneems/schneems/worker_timeout-docs
 - Merge pull request #817 from unleashed/worker-boot-timeout

2.14.0 / 2015-09-18

- 1 minor feature:
 - Make building with SSL support optional
- 1 bug fix:
 - Use Rack::Builder if available. Fixes #735

2.13.4 / 2015-08-16

- 1 bug fix:
 - Use the environment possible set by the config early and from the config file later (if set).

2.13.3 / 2015-08-15

Seriously, I need to revamp config with tests.

- 1 bug fix:
 - Fix preserving options before cleaning for state. Fixes #769

2.13.2 / 2015-08-15

The “clearly I don’t have enough tests for the config” release.

- 1 bug fix:
 - Fix another place binds wasn’t initialized. Fixes #767

2.13.1 / 2015-08-15

- 2 bug fixes:
 - Fix binds being masked in config files. Fixes #765
 - Use options from the config file properly in pumactl. Fixes #764

2.13.0 / 2015-08-14

- 1 minor feature:
 - Add before_fork hooks option.
- 3 bug fixes:
 - Check for OPENSSL_NO_ECDH before using ECDH

- Eliminate logging overhead from JRuby SSL
- Prefer cli options over config file ones. Fixes #669
- 1 deprecation:
 - Add deprecation warning to capistrano.rb. Fixes #673
- 4 PRs merged:
 - Merge pull request #668 from kcollignon/patch-1
 - Merge pull request #754 from nathansamson/before_boot
 - Merge pull request #759 from BenV/fix-centos6-build
 - Merge pull request #761 from lookout/no-log

2.12.3 / 2015-08-03

- 8 minor bugs fixed:
 - Fix Capistrano 'uninitialized constant Puma' error.
 - Fix some ancient and incorrect error handling code
 - Fix uninitialized constant error
 - Remove toplevel rack interspection, require rack on load instead
 - Skip empty parts when chunking
 - Switch from inject to each in config_ru_binds iteration
 - Wrap SSLv3 spec in version guard.
 - ruby 1.8.7 compatibility patches
- 4 PRs merged:
 - Merge pull request #742 from deivid-rodriguez/fix_missing_require
 - Merge pull request #743 from matthewd/skip-empty-chunks
 - Merge pull request #749 from huacnlee/fix-cap-uninitialized-puma-error
 - Merge pull request #751 from costi/compat_1_8_7
- 1 test fix:
 - Add 1.8.7, rbx-1 (allow failures) to Travis.

2.12.2 / 2015-07-17

- 2 bug fix:
 - Pull over and use Rack::URLMap. Fixes #741
 - Stub out peercert on JRuby for now. Fixes #739

2.12.1 / 2015-07-16

- 2 bug fixes:

- Use a constant format. Fixes #737
- Use strerror for Windows sake. Fixes #733
- 1 doc change:
 - typo fix: occured -> occurred
- 1 PR merged:
 - Merge pull request #736 from paulanunda/paulanunda/typo-fix

2.12.0 / 2015-07-14

- 13 bug fixes:
 - Add thread reaping to thread pool
 - Do not automatically use chunked responses when hijacked
 - Do not suppress Content-Length on partial hijack
 - Don't allow any exceptions to terminate a thread
 - Handle ENOTCONN client disconnects when setting REMOTE_ADDR
 - Handle very early exit of cluster mode. Fixes #722
 - Install rack when running tests on travis to use rack/lint
 - Make puma -v and -h return success exit code
 - Make pumactl load config/puma.rb by default
 - Pass options from pumactl properly when pruning. Fixes #694
 - Remove rack dependency. Fixes #705
 - Remove the default Content-Type: text/plain
 - Add Client Side Certificate Auth
- 8 doc/test changes:
 - Added example sourcing of environment vars
 - Added tests for bind configuration on rackup file
 - Fix example config text
 - Update DEPLOYMENT.md
 - Update Readme with example of custom error handler
 - ci: Improve Travis settings
 - ci: Start running tests against JRuby 9k on Travis
 - ci: Convert to container infrastructure for travisci
- 2 ops changes:
 - Check for system-wide rbenv
 - capistrano: Add additional env when start rails
- 16 PRs merged:
 - Merge pull request #686 from jjb/patch-2
 - Merge pull request #693 from rob-murray/update-example-config

- Merge pull request #697 from spk/tests-bind-on-rackup-file
- Merge pull request #699 from deees/fix/require_rack_builder
- Merge pull request #701 from deepj/master
- Merge pull request #702 from Jimdo/thread-reaping
- Merge pull request #703 from deepj/travis
- Merge pull request #704 from grega/master
- Merge pull request #709 from lian/master
- Merge pull request #711 from julik/master
- Merge pull request #712 from yakara-ltd/pumactl-default-config
- Merge pull request #715 from RobotJiang/master
- Merge pull request #725 from rwz/master
- Merge pull request #726 from strenuus/handle-client-disconnect
- Merge pull request #729 from allaire/patch-1
- Merge pull request #730 from iamjarvo/container-infrastructure

2.11.3 / 2015-05-18

- 5 bug fixes:
 - Be sure to unlink tempfiles after a request. Fixes #690
 - Coerce the key to a string before checking. (thar be symbols). Fixes #684
 - Fix hang on bad SSL handshake
 - Remove `enable_sslv3` support from JRuby
- 1 PR merged:
 - Merge pull request #698 from looker/hang-handshake

2.11.2 / 2015-04-11

- 2 minor features:
 - Add `on_worker_fork` hook, which allows to mimic Unicorn's behavior
 - Add shutdown_debug config option
- 4 bug fixes:
 - Fix the Config constants not being available in the DSL. Fixes #683
 - Ignore multiple port declarations
 - Proper 'Connection' header handling compatible with HTTP 1.[01] protocols
 - Use "Puma" instead of "puma" to reporting to New Relic
- 1 doc fixes:
 - Add Gitter badge.
- 6 PRs merged:

- Merge pull request #657 from schneems/schneems/puma-once-port
- Merge pull request #658 from Tomohiro/newrelic-dispatcher-default-update
- Merge pull request #662 from basecrm/connection-compatibility
- Merge pull request #664 from fxposter/on-worker-fork
- Merge pull request #667 from JuanitoFatas/doc/gemspec
- Merge pull request #672 from chulkilee/refactor

2.11.1 / 2015-02-11

- 2 bug fixes:
 - Avoid crash in strange restart conditions
 - Inject the GEM_HOME that bundler into puma-wild's env. Fixes #653
- 2 PRs merged:
 - Merge pull request #644 from bpaquet/master
 - Merge pull request #646 from mkonecny/master

2.11.0 / 2015-01-20

- 9 bug fixes:
 - Add mode as an additional bind option to unix sockets. Fixes #630
 - Advertise HTTPS properly after a hot restart
 - Don't write lowlevel_error_handler to state
 - Fix phased restart with stuck requests
 - Handle spaces in the path properly. Fixes #622
 - Set a default REMOTE_ADDR to avoid using peeraddr on unix sockets. Fixes #583
 - Skip device number checking on jruby. Fixes #586
 - Update extconf.rb to compile correctly on OS X
 - redirect io right after daemonizing so startup errors are shown. Fixes #359
- 6 minor features:
 - Add a configuration option that prevents puma from queueing requests.
 - Add reload_worker_directory
 - Add the ability to pass environment variables to the init script (for Jungle).
 - Add the proctitle tag to the worker. Fixes #633
 - Infer a proctitle tag based on the directory
 - Update lowlevel error message to be more meaningful.
- 10 PRs merged:
 - Merge pull request #478 from rubencaro/master
 - Merge pull request #610 from kwilczynski/master

- Merge pull request #611 from jason1/better-lowlevel-message
- Merge pull request #616 from jc00ke/master
- Merge pull request #623 from raldred/patch-1
- Merge pull request #628 from rdpoor/master
- Merge pull request #634 from deepj/master
- Merge pull request #637 from raskhadafi/patch-1
- Merge pull request #639 from ebeigarts/fix-phased-restarts
- Merge pull request #640 from codehotter/issue-612-dependent-requests-deadlock

2.10.2 / 2014-11-26

- 1 bug fix:
 - Conditionalize thread local cleaning, fixes perf degradation fix
The code to clean out all Thread locals adds pretty significant overhead to a each request, so it has to be turned on explicitly if a user needs it.

2.10.1 / 2014-11-24

- 1 bug fix:
 - Load the app after daemonizing because the app might start threads.

This change means errors loading the app are now reported only in the redirected stdout/stderr.

If you're app has problems starting up, start it without daemon mode initially to test.

2.10.0 / 2014-11-23

- 3 minor features:
 - Added on_worker_shutdown hook mechanism
 - Allow binding to ipv6 addresses for ssl URIs
 - Warn about any threads started during app preload
- 5 bug fixes:
 - Clean out a threads local data before doing work
 - Disable SSLv3. Fixes #591
 - First change the directory to use the correct Gemfile.
 - Only use config.ru binds if specified. Fixes #606
 - Strongish cipher suite with FS support for some browsers
- 2 doc changes:

- Change umask examples to more permissive values
- fix typo in README.md

- 9 Merged PRs:

- Merge pull request #560 from raskhadafi/prune_bundler-bug
- Merge pull request #566 from sheltond/master
- Merge pull request #593 from andruby/patch-1
- Merge pull request #594 from hassox/thread-cleanliness
- Merge pull request #596 from burningTyger/patch-1
- Merge pull request #601 from sorentwo/friendly-umask
- Merge pull request #602 from 1334/patch-1
- Merge pull request #608 from Gu1/master
- Merge remote-tracking branch 'origin/pr/538'

2.9.2 / 2014-10-25

- 8 bug fixes:

- Fix puma-wild handling a restart properly. Fixes #550
- JRuby SSL POODLE update
- Keep deprecated features warnings
- Log the current time when Puma shuts down.
- Fix cross-platform extension library detection
- Use the correct Windows names for OpenSSL.
- Better error logging during startup
- Fixing sexist error messages

- 6 PRs merged:

- Merge pull request #549 from bsnape/log-shutdown-time
- Merge pull request #553 from lowjoel/master
- Merge pull request #568 from mariuz/patch-1
- Merge pull request #578 from danielbuechele/patch-1
- Merge pull request #581 from alexch/slightly-better-logging
- Merge pull request #590 from looker/jruby_disable_sslv3

2.9.1 / 2014-09-05

- 4 bug fixes:

- Cleanup the SSL related structures properly, fixes memory leak
- Fix thread spawning edge case.
- Force a worker check after a worker boots, don't wait 5sec. Fixes #574
- Implement SIGHUP for logs reopening

- 2 PRs merged:

- Merge pull request #561 from theoldreader/sighup
- Merge pull request #570 from havenwood/spawn-thread-edge-case

2.9.0 / 2014-07-12

- 1 minor feature:
 - Add SSL support for JRuby
- 3 bug fixes:
 - Typo BUNDLER_GEMFILE -> BUNDLE_GEMFILE
 - Use fast_write because we can't trust syswrite
 - pumactl - do not modify original ARGV
- 4 doc fixes:
 - BSD-3-Clause over BSD to avoid confusion
 - Deploy doc: clarification of the GIL
 - Fix typo in DEPLOYMENT.md
 - Update README.md
- 6 PRs merged:
 - Merge pull request #520 from misfo/patch-2
 - Merge pull request #530 from lookout/jruby-ssl
 - Merge pull request #537 from vlmonk/patch-1
 - Merge pull request #540 from allaire/patch-1
 - Merge pull request #544 from chulkilee/bsd-3-clause
 - Merge pull request #551 from jcxplorer/patch-1

2.8.2 / 2014-04-12

- 4 bug fixes:
 - During upgrade, change directory in main process instead of workers.
 - Close the client properly on error
 - Capistrano: fallback from phased restart to start when not started
 - Allow tag option in conf file
- 4 doc fixes:
 - Fix Puma daemon service README typo
 - `preload_app!` instead of `preload_app`
 - add preload_app and prune_bundler to example config
 - allow changing of worker_timeout in config file
- 11 PRs merged:

- Merge pull request #487 from ckuttruff/master
- Merge pull request #492 from ckuttruff/master
- Merge pull request #493 from alepore/config_tag
- Merge pull request #503 from mariuz/patch-1
- Merge pull request #505 from sammcj/patch-1
- Merge pull request #506 from FlavourSys/config_worker_timeout
- Merge pull request #510 from momer/rescue-block-handle-servers-fix
- Merge pull request #511 from macool/patch-1
- Merge pull request #514 from edogawaconan/refactor_env
- Merge pull request #517 from misfo/patch-1
- Merge pull request #518 from LongMan/master

2.8.1 / 2014-03-06

- 1 bug fixes:
 - Run puma-wild with proper deps for prune_bundler
- 2 doc changes:
 - Described the configuration file finding behavior added in 2.8.0 and how to disable it.
 - Start the deployment doc
- 6 PRs merged:
 - Merge pull request #471 from arthurnn/fix_test
 - Merge pull request #485 from joneslee85/patch-9
 - Merge pull request #486 from joshwlewis/patch-1
 - Merge pull request #490 from tobinibot/patch-1
 - Merge pull request #491 from brianknight10/clarify-no-config

2.8.0 / 2014-02-28

- 8 minor features:
 - Add ability to autoload a config file. Fixes #438
 - Add ability to detect and terminate hung workers. Fixes #333
 - Add booted_workers to stats response
 - Add config to customize the default error message
 - Add prune_bundler option
 - Add worker indexes, expose them via on_worker_boot. Fixes #440
 - Add pretty process name
 - Show the ruby version in use
- 7 bug fixes:

- Added 408 status on timeout.
- Be more hostile with sockets that write block. Fixes #449
- Expect at_exit to exclusively remove the pidfile. Fixes #444
- Expose latency and listen backlog via bind query. Fixes #370
- JRuby raises IOError if the socket is there. Fixes #377
- Process requests fairly. Fixes #406
- Rescue SystemCallError as well. Fixes #425
- 4 doc changes:
 - Add 2.1.0 to the matrix
 - Add Code Climate badge to README
 - Create signals.md
 - Set the license to BSD. Fixes #432
- 14 PRs merged:
 - Merge pull request #428 from alexeyfrank/capistrano_default_hooks
 - Merge pull request #429 from namusyaka/revert-const_defined
 - Merge pull request #431 from mrb/master
 - Merge pull request #433 from alepore/process-name
 - Merge pull request #437 from ibrahima/master
 - Merge pull request #446 from sudara/master
 - Merge pull request #451 from pwiebe/status_408
 - Merge pull request #453 from joevandyk/patch-1
 - Merge pull request #470 from arthurnn/fix_458
 - Merge pull request #472 from rubencaro/master
 - Merge pull request #480 from jjb/docs-on-running-test-suite
 - Merge pull request #481 from schneems/master
 - Merge pull request #482 from prathamesh-sonpatki/signals-doc-cleanup
 - Merge pull request #483 from YotpoLtd/master

2.7.1 / 2013-12-05

- 1 bug fix:
 - Keep STDOUT/STDERR the right mode. Fixes #422

2.7.0 / 2013-12-03

- 1 minor feature:
 - Adding TTIN and TTOU to increment/decrement workers
- N bug fixes:
 - Always use our Process.daemon because it's not busted

- Add capistrano restart failback to start.
- Change position of `cd` so that rvm gemset is loaded
- Clarify some platform specifics
- Do not close the pipe sockets when retrying
- Fix String#byteslice for Ruby 1.9.1, 1.9.2
- Fix compatibility with 1.8.7.
- Handle IOError closed stream in IO.select
- Increase the max URI path length to 2048 chars from 1024 chars
- Upstart jungle use config/puma.rb instead

2.6.0 / 2013-09-13

- 2 minor features:
 - Add support for event hooks
 - ** Add a hook for state transitions
 - Add phased restart to capistrano recipe.
- 4 bug fixes:
 - Convince workers to stop by SIGKILL after timeout
 - Define RSTRING_NOT_MODIFIED for Rubinius performance
 - Handle BrokenPipe, StandardError and IOError in fat_wrote and break out
 - Return success status to the invoking environment

2.5.1 / 2013-08-13

- 2 bug fixes:
 - Keep jruby daemon mode from retrying on a hot restart
 - Extract version from const.rb in gemspec

2.5.0 / 2013-08-08

- 2 minor features:
 - Allow configuring pumactl with config.rb
 - make `pumactl restart` start puma if not running
- 6 bug fixes:
 - Autodetect ruby managers and home directory in upstart script
 - Convert header values to string before sending.
 - Correctly report phased-restart availability
 - Fix pidfile creation/deletion race on jruby daemonization
 - Use integers when comparing thread counts
 - Fix typo in using lopez express (raw tcp) mode

- 6 misc changes:
 - Fix typo in phased-restart response
 - Uncomment setuid/setgid by default in upstart
 - Use Puma: ☐:PUMA_VERSION in gemspec
 - Update upstart comments to reflect new commandline
 - Remove obsolete pumactl instructions; refer to pumactl for details
 - Make Bundler used puma.gemspec version agnostic

2.4.1 / 2013-08-07

- 1 experimental feature:
 - Support raw tcp servers (aka Lopez Express mode)

2.4.0 / 2013-07-22

- 5 minor features:
 - Add PUMA_JRUBY_DAEMON_OPTS to get around agent starting twice
 - Add ability to drain accept socket on shutdown
 - Add port to DSL
 - Adds support for using puma config file in capistrano deploys.
 - Make phased_restart fallback to restart if not available
- 10 bug fixes:
 - Be sure to only delete the pid in the master. Fixes #334
 - Call out -C/--config flags
 - Change parser symbol names to avoid clash. Fixes #179
 - Convert thread pool sizes to integers
 - Detect when the jruby daemon child doesn't start properly
 - Fix typo in CLI help
 - Improve the logging output when hijack is used. Fixes #332
 - Remove unnecessary thread pool size conversions
 - Setup :worker_boot as an Array. Fixes #317
 - Use 127.0.0.1 as REMOTE_ADDR of unix client. Fixes #309

2.3.2 / 2013-07-08

- 1 bug fix:
 - Move starting control server to after daemonization.

2.3.1 / 2013-07-06

- 2 bug fixes:
 - Include the right files in the Manifest.
 - Disable inheriting connections on restart on windows. Fixes #166
- 1 doc change:
 - Better document some platform constraints

2.3.0 / 2013-07-05

- 1 major bug fix:
 - Stabilize control server, add support in cluster mode
- 5 minor bug fixes:
 - Add ability to cleanup stale unix sockets
 - Check status data better. Fixes #292
 - Convert raw IO errors to ConnectionError. Fixes #274
 - Fix sending Content-Type and Content-Length for no body status. Fixes #304
 - Pass state path through to `pumactl start`. Fixes #287
- 2 internal changes:
 - Refactored modes into separate classes that CLI uses
 - Changed CLI to take an Events object instead of stdout/stderr (API change)

2.2.2 / 2013-07-02

- 1 bug fix:
 - Fix restart_command in the config

2.2.1 / 2013-07-02

- 1 minor feature:
 - Introduce preload flag
- 1 bug fix:
 - Pass custom restart command in JRuby

2.2.0 / 2013-07-01

- 1 major feature:

- Add ability to preload rack app
- 2 minor bugfixes:
 - Don't leak info when not in development. Fixes #256
 - Load the app, then bind the ports

2.1.1 / 2013-06-20

- 2 minor bug fixes:
 - Fix daemonization on jruby
 - Load the application before daemonizing. Fixes #285

2.1.0 / 2013-06-18

- 3 minor features:
 - Allow listening socket to be configured via Capistrano variable
 - Output results from 'stat's command when using pumactl
 - Support systemd socket activation
- 15 bug fixes:
 - Deal with pipes closing while stopping. Fixes #270
 - Error out early if there is no app configured
 - Handle ConnectionError rather than the lowlevel exceptions
 - tune with `-C` config file and `on_worker_boot`
 - use `-w`
 - Fixed some typos in upstart scripts
 - Make sure to use bytesize instead of size (MiniSSL write)
 - Fix an error in puma-manager.conf
 - fix: stop leaking sockets on restart (affects ruby 1.9.3 or before)
 - Ignore errors on the cross-thread pipe. Fixes #246
 - Ignore errors while uncorking the socket (it might already be closed)
 - Ignore the body on a HEAD request. Fixes #278
 - Handle all engine data when possible. Fixes #251.
 - Handle all read exceptions properly. Fixes #252
 - Handle errors from the server better
- 3 doc changes:
 - Add note about on_worker_boot hook
 - Add some documentation for Clustered mode
 - Added quotes to /etc/puma.conf

2.0.1 / 2013-04-30

- 1 bug fix:
 - Fix not starting on JRuby properly

2.0.0 / 2013-04-29

RailsConf 2013 edition!

- 2 doc changes:
 - Start with rackup -s Puma, NOT rackup -s puma.
 - Minor doc fixes in the README.md, Capistrano section
- 2 bug fixes:
 - Fix reading RACK_ENV properly. Fixes #234
 - Make cap recipe handle tmp/sockets; fixes #228
- 3 minor changes:
 - Fix capistrano recipe
 - Fix stdout/stderr logs to sync outputs
 - allow binding to IPv6 addresses

2.0.0.b7 / 2013-03-18

- 5 minor enhancements:
 - Add -q option for :start
 - Add -V, --version
 - Add default Rack handler helper
 - Upstart support
 - Set worker directory from configuration file
- 12 bug fixes:
 - Close the binder in the right place. Fixes #192
 - Handle early term in workers. Fixes #206
 - Make sure that the default port is 80 when the request doesn't include HTTP_X_FORWARDED_PROTO.
 - Prevent Errno::EBADF errors on restart when running ruby 2.0
 - Record the proper `@master_pid`
 - Respect the header HTTP_X_FORWARDED_PROTO when the host doesn't include a port number.
 - Retry EAGAIN/EWOULDBLOCK during syswrite
 - Run exec properly to restart. Fixes #154
 - Set Rack run_once to false
 - Synchronize all access to `@timeouts`. Fixes #208

- Write out the state post-daemonize. Fixes #189
- Prevent crash when all workers are gone

2.0.0.b6 / 2013-02-06

- 2 minor enhancements:
 - Add hook for running when a worker boots
 - Advertise the Configuration object for apps to use.
- 1 bug fix:
 - Change directory in working during upgrade. Fixes #185

2.0.0.b5 / 2013-02-05

- 2 major features:
 - Add phased worker upgrade
 - Add support for the rack hijack protocol
- 2 minor features:
 - Add -R to specify the restart command
 - Add config file option to specify the restart command
- 5 bug fixes:
 - Cleanup pipes properly. Fixes #182
 - Daemonize earlier so that we don't lose app threads. Fixes #183
 - Drain the notification pipe. Fixes #176, thanks [@cryo28](#)
 - Move write_pid to after we daemonize. Fixes #180
 - Redirect IO properly and emit message for checkpointing

2.0.0.b4 / 2012-12-12

- 4 bug fixes:
 - Properly check #syswrite's value for variable sized buffers. Fixes #170
 - Shutdown status server properly
 - Handle char vs byte and mixing syswrite with write properly
 - made MiniSSL validate key/cert file existence

2.0.0.b3 / 2012-11-22

- 1 bug fix:
 - Package right files in gem

2.0.0.b2 / 2012-11-18

- 5 minor feature:
 - Now Puma is bundled with an capistrano recipe. Just require `'puma/capistrano'` in you `deploy.rb`
 - Only inject CommonLogger in development mode
 - Add `-p` option to `pumactl`
 - Add ability to use `pumactl` to start a server
 - Add options to daemonize puma
- 7 bug fixes:
 - Reset the IOBuffer properly. Fixes #148
 - Shutdown gracefully on JRuby with Ctrl-C
 - Various methods to get newrelic to start. Fixes #128
 - fixing syntax error at capistrano recipe
 - Force ECONNRESET when read returns nil
 - Be sure to empty the drain the todo before shutting down. Fixes #155
 - allow for alternate locations for status app

2.0.0.b1 / 2012-09-11

- 1 major feature:
 - Optional worker process mode (`-w`) to allow for process scaling in addition to thread scaling
- 1 bug fix:
 - Introduce `Puma::MiniSSL` to be able to properly control doing nonblocking SSL

NOTE: SSL support in JRuby is not supported at present. Support will be added back in a future date when a java `Puma::MiniSSL` is added.

1.6.3 / 2012-09-04

- 1 bug fix:
 - Close sockets waiting in the reactor when a hot restart is performed so that browsers reconnect on the next request

1.6.2 / 2012-08-27

- 1 bug fix:
 - Rescue `StandardError` instead of `IOError` to handle `SystemCallErrors`

as well as other application exceptions inside the reactor.

1.6.1 / 2012-07-23

- 1 packaging bug fixed:
 - Include missing files

1.6.0 / 2012-07-23

- 1 major bug fix:
 - Prevent slow clients from starving the server by introducing a dedicated IO reactor thread. Credit for reporting goes to [@meh](#).

1.5.0 / 2012-07-19

- 7 contributors to this release:
 - Christian Mayer
 - Darío Javier Cravero
 - Dirkjan Bussink
 - Gianluca Padovani
 - Santiago Pastorino
 - Thibault Jouan
 - tomykaira
- 6 bug fixes:
 - Define RSTRING_NOT_MODIFIED for Rubinius
 - Convert status to integer. Fixes #123
 - Delete pidfile when stopping the server
 - Allow compilation with -Werror=format-security option
 - Fix wrong HTTP version for a HTTP/1.0 request
 - Use String#bytesize instead of String#length
- 3 minor features:
 - Added support for setting RACK_ENV via the CLI, config file, and rack app
 - Allow Server#run to run sync. Fixes #111
 - Puma can now run on windows

1.4.0 / 2012-06-04

- 1 bug fix:
 - SCRIPT_NAME should be passed from env to allow mounting apps

- 1 experimental feature:
 - Add puma.socket key for direct socket access

1.3.1 / 2012-05-15

- 2 bug fixes:
 - use #bytesize instead of #length for Content-Length header
 - Use StringIO properly. Fixes #98

1.3.0 / 2012-05-08

- 2 minor features:
 - Return valid Rack responses (passes Lint) from status server
 - Add -I option to specify \$LOAD_PATH directories
- 4 bug fixes:
 - Don't join the server thread inside the signal handle. Fixes #94
 - Make NullIO#read mimic IO#read
 - Only stop the status server if it's started. Fixes #84
 - Set RACK_ENV early in cli also. Fixes #78
- 1 new contributor:
 - Jesse Cooke

1.2.2 / 2012-04-28

- 4 bug fixes:
 - Report a lowlevel error to stderr
 - Set a fallback SERVER_NAME and SERVER_PORT
 - Keep the encoding of the body correct. Fixes #79
 - show error.to_s along with backtrace for low-level error

1.2.1 / 2012-04-11

1 bug fix:

- Fix rack.url_scheme for SSL servers. Fixes #65

1.2.0 / 2012-04-11

1 major feature:

- When possible, the internal restart does a “hot restart” meaning the server sockets remains open, so no connections are lost.

1 minor feature:

- More helpful fallback error message

6 bug fixes:

- Pass the proper args to unknown_error. Fixes #54, #58
- Stop the control server before restarting. Fixes #61
- Fix reporting https only on a true SSL connection
- Set the default content type to 'text/plain'. Fixes #63
- Use REUSEADDR. Fixes #60
- Shutdown gracefully on SIGTERM. Fixes #53

2 new contributors:

- Seamus Abshire
- Steve Richert

1.1.1 / 2012-03-30

1 bugfix:

- Include puma/compat.rb in the gem (oops!)

1.1.0 / 2012-03-30

1 bugfix:

- Make sure that the unix socket has the perms 0777 by default

1 minor feature:

- Add umask param to the unix:// bind to set the umask

1.0.0 / 2012-03-29

- Released!

Release

Before Release

- Make sure tests pass and your last local commit matches master.
- Run tests with latest jruby
- Update the version in `const.rb`.
- Make sure there is a history entry in `History.md`.
- On minor version updates i.e. from 3.10.x to 3.11.x update the “codename” in `const.rb`.

Release process

Using “3.7.1” as a version example.

1. `bundle exec rake release`
2. Switch to latest JRuby version
3. `rake java gem`
4. `gem push pkg/puma-3.7.1-java.gem`