

# **Ecmo**

Easy Configuration Management  
Orchestration

[Logout](#)

Logged in as airesgabriel/other

# Ecmo

Easy Configuration Management Orchestration

[Dashboard](#)[Dash\(current\)](#)[Available Jobs](#)[Host Info](#)[Boot Time](#)[System Load](#)[Memory Usage](#)[Net I/O](#)[Disk Usage](#)[Partitions](#)[Processes](#)

Built with S2 and Spider-Gazelle

[Check the Github Repo](#)

# **Por que outra ferramenta de gestão de configuração?**

# Comparativo com ferramentas populares

## Puppet

- + Já utilizado na Anatel
- + Idempotente
- - Versão utilizada é muito antiga
- - Topologia cliente/servidor, requer gestão centralizada, instalação de repositórios e pacotes adicionais no cliente
- - Linguagem de configuração própria, conhecimentos não transferíveis de outros domínios
- - Necessidade de hacks para uso standalone

## Ecmo

- + Idempotente
- + Instalação standalone a partir de um binário autocontido
- + Utiliza ruby como DSL

# Comparativo com ferramentas populares

## Ansible

- + Agentless (utiliza sessões ssh)
- - Não idempotente
- - Playbooks em YAML, menor flexibilidade
- - Playbooks tendem a ser lentos devido ao tráfego de dados pela rede, operações realizadas em hosts distintos, muitas sessões ssh, etc

## Ecmo

- + Utiliza agente, porém é autocontido (não requer quaisquer dependências ou repositórios de terceiros)
- + Idempotente
- + A linguagem ruby permite maior flexibilidade na definição das configurações
- + Execução local permite a conclusão de jobs em menor tempo

# Comparação com ferramentas populares

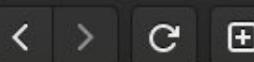
## Chef

- + Flexível, utiliza Ruby como DSL
- - Arquitetura cliente/servidor requer infraestrutura mais complexa, além da instalação de repositórios e pacotes adicionais no cliente
- - Não idempotente

## Ecmo

- + Utiliza Ruby como DSL
- + Standalone, não requer a instalação de pacotes adicionais.
- + Utiliza sintaxe semelhante ao Chef, porém idempotente, graças ao utilitário mitamae, que embarca um interpretador mruby estaticamente compilado

# Instalação



Search or jump to...

Pull requests Issues Marketplace Explore



gabriel-aires / emco

Public

Unpin

Unwatch 2

Fork 0

Star 0

Code

Issues

Pull requests

Actions

Projects

Wiki

Security

Insights

Settings

main

2 branches

12 tags

Go to file

Add file

Code



gabriel-aires added authorization management/logging, updated ser... ... 2162da9 5 days ago 70 commits

dist	added authorization management/logging, updated service c...	5 days ago
spec	project based on spider-gazelle template	3 months ago
src	added authorization management/logging, updated service c...	5 days ago
vfs	added authorization management/logging, updated service c...	5 days ago
.gitattributes	project based on spider-gazelle template	3 months ago
.gitignore	fix service configuration, better logging	2 months ago
LICENSE	project based on spider-gazelle template	3 months ago
README.md	project renamed, added notice and flash functionality, todo: fi...	6 days ago
Vagrantfile	project renamed, added notice and flash functionality, todo: fi...	6 days ago
shard.lock	libssh2 statically linked, added fake authorization methods	8 days ago

## About

Easy Configuration Management  
Orchestration

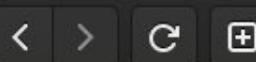
- Readme
- MPL-2.0 License
- 0 stars
- 2 watching
- 0 forks

## Releases 7

 v0.9.0 Latest  
5 days ago

+ 6 releases

## Packages



Search or jump to...

Pull requests Issues Marketplace Explore

gabriel-aires / **ecmo** Public

Unpin

Unwatch 2

Fork 0

Star 0

Code

Issues

Pull requests

Actions

Projects

Wiki

Security

Insights

Settings

Releases

Tags

Draft a new release

Find a release

5 days ago



gabriel-aires



v0.9.0



2162da9

Compare ▾

**v0.9.0**

Latest



- Added user authentication/authorization/logging
- Improved theming and layout
- Bugfixes

## ▼ Assets 3

[ecmo-linux-x64.bin](#)

19.5 MB

[Source code \(zip\)](#) [Source code \(tar.gz\)](#)



Pasta pessoal

Public



Recentes

Favorito

Pasta pessoal

Documents

Downloads

Music

Pictures

Videos

Lixeira

Public

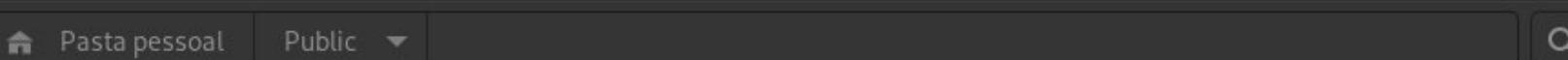
Templates

Outros locais

Nome ▾ Tamanho Modificado

ecmo-linux-x64.bin 20.5 MB 22:08 ⭐

- Nova pasta Shift+Ctrl+N
- Adicionar marcador Ctrl+D
- Colar
- Selecionar tudo Ctrl+A
- Abrir no terminal
- Propriedades



Recentes

Favorito

Pasta pessoal

Documents

Downloads

Music

Pictures

Videos

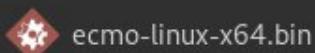
Lixeira

Public

Templates

Outros locais

Nome



Tamanho

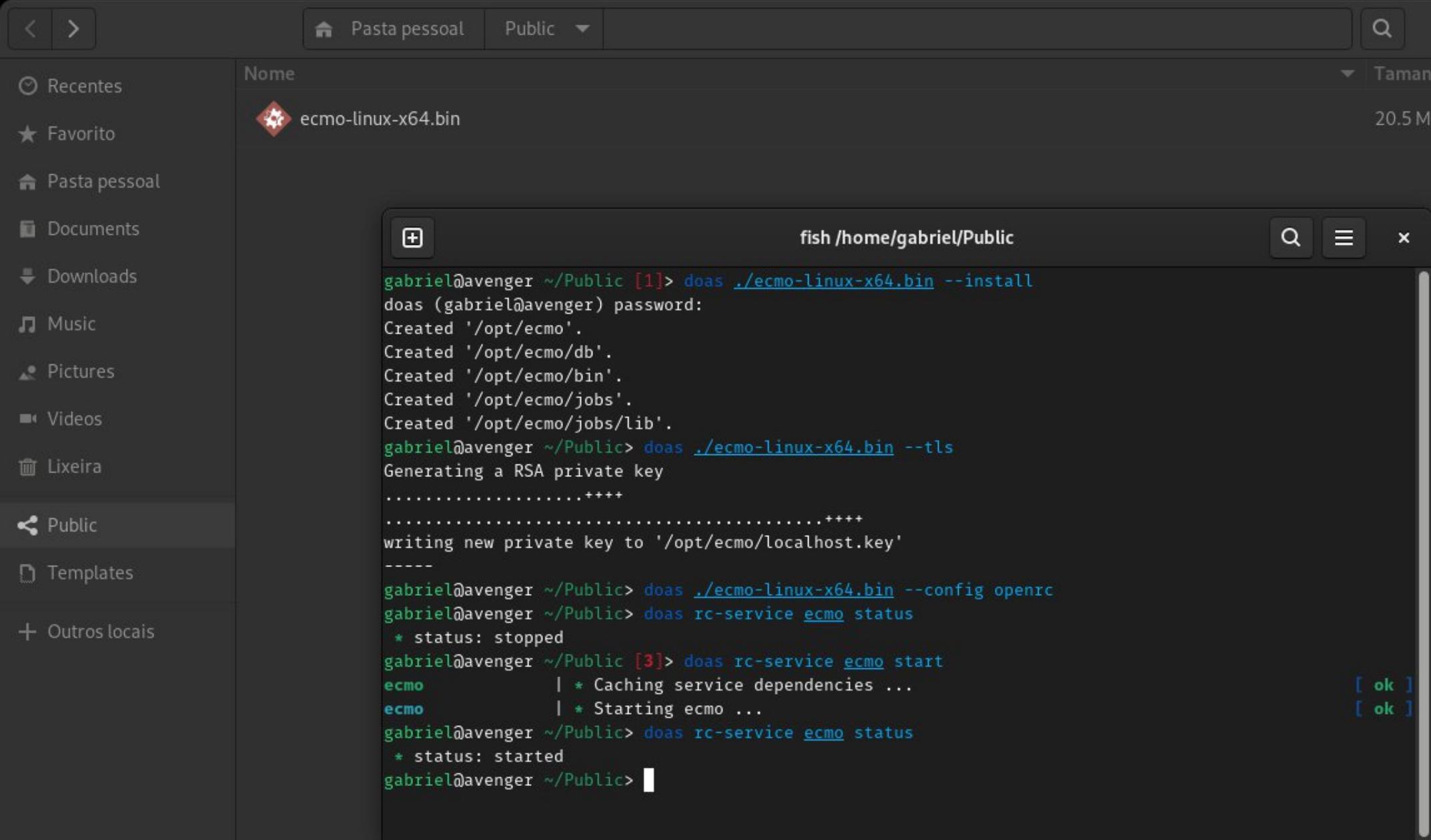
20.5 MB

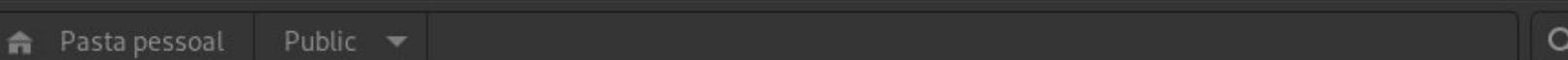
Modificado

22:08



```
fish /home/gabriel/Public
+
Welcome to fish, the friendly interactive shell
Type help for instructions on how to use fish
gabriel@avenger ~/Public> ls -l ecmo-linux-x64.bin
-rw-r--r-- 1 gabriel gabriel 20496232 Feb 15 22:08 ecmo-linux-x64.bin
gabriel@avenger ~/Public> chmod +x ecmo-linux-x64.bin
gabriel@avenger ~/Public> ls -l ecmo-linux-x64.bin
-rwxr-xr-x 1 gabriel gabriel 20496232 Feb 15 22:08 ecmo-linux-x64.bin
gabriel@avenger ~/Public> ./ecmo-linux-x64.bin -h
Usage: ./ecmo-linux-x64.bin [arguments]
      -h, --help          Show this help
      -v, --version        Display the application version
      -i, --install        Install database, binaries and job definitions
      -t, --tls            Generate temporary self-signed certificate (requires openssl)
      -r, --routes         List the application routes
      -g URL, --get=URL   Perform a basic health check by requesting the URL
      -b HOST, --bind=HOST Specifies the server host
      -p PORT, --port=PORT Specifies the server port
      -w COUNT, --workers=COUNT Specifies the number of processes to handle requests
      -c INIT, --config=INIT Configure init service: openrc|systemd
gabriel@avenger ~/Public>
```





Recentes

Favorito

Pasta pessoal

Documents

Downloads

Music

Pictures

Videos

Lixeira

Public

Templates

Outros locais

Nome	Tamanho	Modificado
------	---------	------------

ecmo-linux-x64.bin	20.5 MB	22:08
--------------------	---------	-------

```
[+] fish /home/gabriel/Public
gabriel@avenger ~/Public> doas cat /opt/ecmo/service.log
Launching Ecmo v0.9
Calling job "db_cleaner" (cron 40 * * * * *)
Calling job "disk_monitor" (cron 5 * * * * *)
Calling job "host_monitor" (cron 8 * * * * *)
Calling job "itamae_scheduler" (in 00:00:10)
Calling job "job_discovery" (in 00:00:05)
Calling job "load_monitor" (cron 10 * * * * *)
Calling job "memory_monitor" (cron 12 * * * * *)
Calling job "net_monitor" (cron 15 * * * * *)
Calling job "partition_monitor" (cron 17 * * * * *)
Calling job "pid_monitor" (cron 20 * * * * *)
Listening on https://127.0.0.1:3000
Calling job "host_inventory" (cron * * * * *)
Calling job "app_restart" (on demand)
Calling job "host_report" (cron */5 * * * * *)
Calling job "app_shutdown" (on demand)
gabriel@avenger ~/Public>
```

## Ecmo: Easy Configuration Management Orchestration

 Username PasswordLogin

Built with S2 and Spider-Gazelle

Check the Github Repo

## Exemplo de instalação em RedHat

```
wget https://github.com/gabriel-  
aires/ecmo/releases/download/v0.9.0/ecmo-linux-  
x64.bin  
chmod +x ecmo-linux-x64.bin  
sudo ./ecmo-linux-x64.bin -i  
sudo ./ecmo-linux-x64.bin -t  
sudo ./ecmo-linux-x64.bin -c systemd  
sudo systemctl enable ecmo  
sudo systemctl start ecmo
```

# Configuração

fish /home/gabriel

```
gabriel@avenger ~> doas cat /etc/conf.d/ecmo
doas (gabriel@avenger) password:
MODE="production"
SERVER_PORT=3000
SERVER_HOST="127.0.0.1"
DB_RETENTION=7
SESSION_KEY="_ecmo_"
SESSION_SECRET="4f74c0b358d5bab4000dd3c75465dc2c"
ALLOW_READ=""
ALLOW_WRITE="wheel"
gabriel@avenger ~>
```

# VARIÁVEIS DE AMBIENTE

OpenRC : /etc/conf.d/ecmo

systemd: /opt/ecmo/ecmo.conf

---

```
MODE="production"                                # ambiente
SERVER_PORT=3000                                 # porta https
SERVER_HOST="127.0.0.1"                           # IP servidor
DB_RETENTION=7                                    # métricas/logs
SESSION_KEY="_ecmo_"                             # chave cookie
SESSION_SECRET="4f74c0b358d5bab4000dd3c75465dc2c" # assinatura
ALLOW_READ=""                                     # grupo leitura
ALLOW_WRITE="wheel"                              # grupo escrita
```

fish /home/gabriel

```
gabriel@avenger ~> doas exa --long --tree --sort-type /opt/ecmo
drwxr-xr-x  - root 15 Feb 22:11 /opt/ecmo
drwxr-xr-x  - root 15 Feb 22:10
.rwXr-xr-x  20M root 15 Feb 22:10
.rwXr-xr-x  2.0M root 15 Feb 22:10
drwxr-xr-x  - root 15 Feb 22:11
.rw-r--r--  451k root 15 Feb 22:22
.rw-r--r--   33k root 15 Feb 22:24
.rw-r--r--  4.3M root 15 Feb 22:24
drwxr-xr-x  - root 15 Feb 22:10
drwxr-xr-x  - root 15 Feb 22:10
.rw-r--r--   408 root 15 Feb 22:10
.rw-r--r--   135 root 15 Feb 22:10
.rw-r--r--   133 root 15 Feb 22:10
.rw-r--r--   165 root 15 Feb 22:10
.rw-r--r--   164 root 15 Feb 22:10
.rw-r--r--   12k root 15 Feb 22:16
.rw-----  3.0k root 15 Feb 22:14
.rw-----  1.8k root 15 Feb 22:10
.rw-----  3.3k root 15 Feb 22:10
.rw-----  6.0k root 15 Feb 22:15
gabriel@avenger ~>
```

The terminal window shows a command-line interface using the fish shell. The user is navigating through a directory structure under /opt/ecmo. The command used is doas exa --long --tree --sort-type, which provides a detailed listing of files and their metadata. The listing includes file names, permissions (e.g., drwxr-xr-x), sizes (e.g., 20M, 2.0M, 451k, etc.), owners (root), and modification times (e.g., 15 Feb 22:11, 22:10, etc.). The terminal window has a dark theme and is set against a background image of the Parthenon at night.

# ARQUIVOS

---

– Serviço	—
/etc/init.d/ecmo	# Configuração serviço (OpenRC)
/etc/systemd/system/ecmo.service	# Configuração serviço (systemd)
– Instalação	—
/opt/ecmo/	# Raiz da instalação
/opt/ecmo/bin	# Binários ecmo/mitamae
/opt/ecmo/db	# Banco de dados embarcado SQLite
/opt/ecmo/*.log	# Logs de acesso, erro, serviço
/opt/ecmo/localhost.(cert key)	# Certificado e chave TLS (HTTPS)
– Jobs	—
/opt/ecmo/jobs/*.rb	# Jobs de configuração
/opt/ecmo/jobs/lib/*.rb	# Classes e scripts auxiliares
/opt/ecmo/jobs/lib/**/files	# Configurações estáticas do host
/opt/ecmo/jobs/lib/**/templates	# Configurações dinâmicas do host

---

# Autenticação / Autorização

# Autenticação

- Utiliza usuário local ou de rede com acesso SSH ao host.
- Usuários de rede devem ter um UID local já mapeado via SSSD
- UIDs devem ser únicos para evitar problemas de segurança.  
Embora o padrão POSIX permita UIDs (assim como GIDs) compartilhados entre contas, a prática não é recomendada.
- Documentação auxiliar da Red Hat:
  - [Configuração do SSSD](#)
  - [Gestão de credenciais](#)

# Autorização

- As variáveis ALLOW\_READ e ALLOW\_WRITE definem listas de grupos (locais ou de rede) com permissão de leitura (dashboards, consumo de métricas, visualização de logs) ou escrita (disparo de jobs + edição de configurações)
- O permissionamento é checado no backend por filtros de autorização executados antes do handler associado a uma rota em particular.
  - Rotas para login e logout são públicas.
  - Rotas GET para ações #index e #show exigem acesso de leitura.
  - Via de regra, as demais rotas exigem acesso de escrita.

# Rotas RESTful (exemplo)

The methods defined by the `MyPhotos` class define the routes that will be created. These are the routes that will be auto-generated if you define the appropriate method.

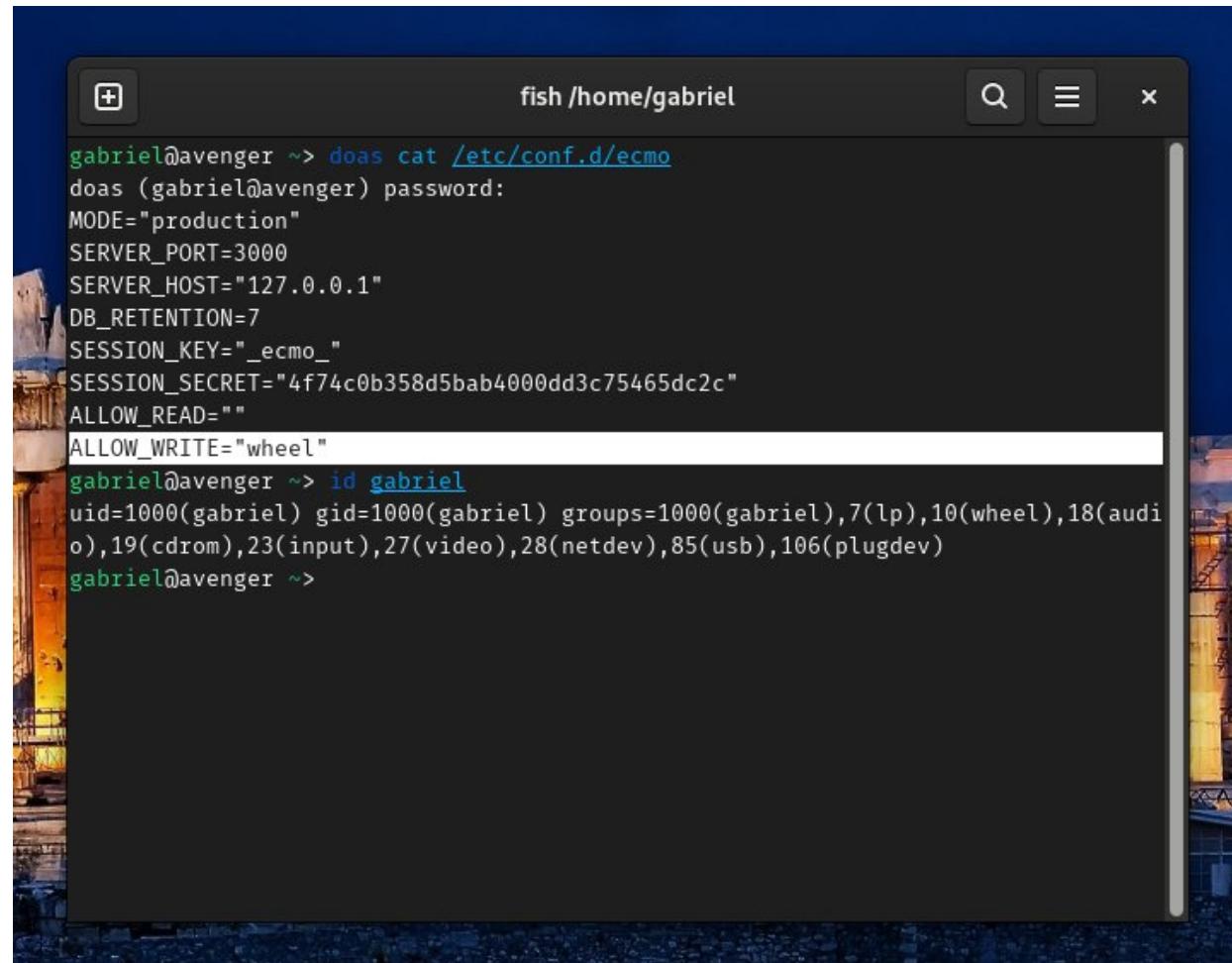
HTTP Verb	Path	Controller#Action	Used for
GET	<code>/my_photos/</code>	<code>my_photos#index</code>	display a list of all your photos
GET	<code>/my_photos/new</code>	<code>my_photos#new</code>	return an HTML form for creating a new photo
POST	<code>/my_photos/</code>	<code>my_photos#create</code>	create a new photo
GET	<code>/my_photos/:id</code>	<code>my_photos#show</code>	display a specific photo
GET	<code>/my_photos/:id/edit</code>	<code>my_photos#edit</code>	return an HTML form for editing a photo
PATCH	<code>/my_photos/:id</code>	<code>my_photos#update</code>	update a specific photo (possibly meta data)
PUT	<code>/my_photos/:id</code>	<code>my_photos#replace</code>	replace a specific photo
DELETE	<code>/my_photos/:id</code>	<code>my_photos#destroy</code>	delete a specific photo

Because the router uses the HTTP verb and URL to match inbound requests, four URLs map to eight different actions.

# Rotas Ecmo

Método HTTP	Padrão de URI	Recurso / Ação	Descrição	Acesso
GET	/	Home#index	Página inicial	Leitura
GET	/assets/:id	Assets#show	Arquivos estáticos: img,js,css	Leitura
GET	/dashboard	Dashboard#index	Painel de monitoramento	Leitura
GET	/dashboard/realtime	Dashboard#realtime	Painel instantâneo	Leitura
GET	/jobs	Jobs#index	Lista de jobs	Leitura
GET	/jobs/:id	Jobs#show	Log do job	Leitura
GET	/metrics/:id	Metrics#show	Endpoint de métricas (JSON)	Leitura
GET	/sessions/new	Sessions#new	Formulário de login	Público
POST	/sessions	Sessions#create	Autenticação	Público
DELETE	/sessions/:id	Sessions#destroy	Logout	Público
GET	/*/new, /*/:id/edit	*#new, *#edit	Formulários de configuração	Escrita
POST, PATCH, PUT, DELETE	/*, /*/:id	*#create, *#update, *#replace, *#destroy	Disparo de jobs, edição de configurações, etc	Escrita

# Grupo de escrita padrão: wheel



A screenshot of a terminal window titled "fish /home/gabriel". The terminal displays the following text:

```
gabriel@avenger ~> doas cat /etc/conf.d/ecmo
doas (gabriel@avenger) password:
MODE="production"
SERVER_PORT=3000
SERVER_HOST="127.0.0.1"
DB_RETENTION=7
SESSION_KEY="_ecmo_"
SESSION_SECRET="4f74c0b358d5bab4000dd3c75465dc2c"
ALLOW_READ=""
ALLOW_WRITE="wheel"
gabriel@avenger ~> id gabriel
uid=1000(gabriel) gid=1000(gabriel) groups=1000(gabriel),7(lp),10(wheel),18(audio),19(cdrom),23(input),27(video),28(netdev),85(usb),106(plugdev)
gabriel@avenger ~>
```

## Padrão de permissionamento

Por padrão, a ferramenta não define quaisquer grupos para acesso de leitura, sendo restrita a uso administrativo.

Para atribuir um ou mais grupos de leitura e escrita, edite os parâmetros “ALLOW\_READ” e “ALLOW\_WRITE” utilizando nomes de grupo separados por espaço. Quando houver mais de um grupo, utilize aspas para delimitar o valor do campo

Ex: ALLOW\_WRITE=“wheel sudo admins.linux.sas.pd”

# Jobs

**Configuração, execução e logs**



Logout

Logged in as gabriel/regular

# Jobs

## System Configuration

### Scheduled Jobs

▶ Host Inventory

▶ Host Report

### On-Demand Jobs

▶ App Restart

▶ App Shutdown

# Exemplo

▼ Host Report

Schedule	*/5 * * * *
Path	/opt/ecmo/jobs/host_report.rb
Definition	<pre># name:          Host Report # schedule:     */5 * * * * # log:           debug # revision:     1  include_recipe "lib/host.rb"  puts Host.inventory(node).to_yaml</pre>
Version	1
Log level	Debug
Last Run	<a href="#">View Log</a>

fish /home/gabriel

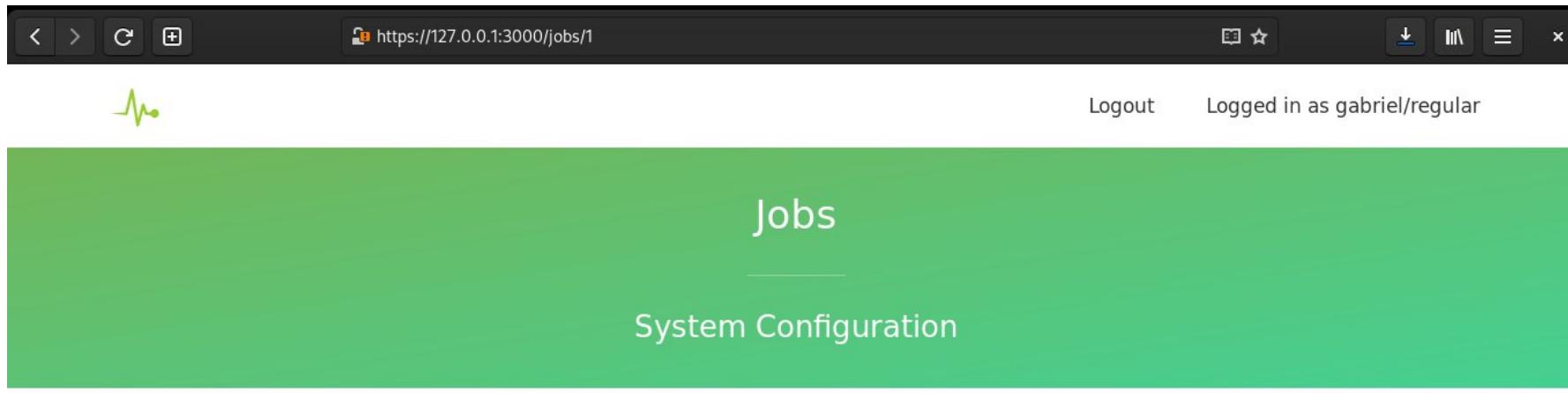
```
gabriel@avenger ~> doas exa --long --tree --sort-type /opt/ecmo
drwxr-xr-x  - root 15 Feb 22:11 /opt/ecmo
drwxr-xr-x  - root 15 Feb 22:10 .ecmo
drwxr-xr-x 20M root 15 Feb 22:10 .mitamae
drwxr-xr-x 2.0M root 15 Feb 22:10 db
drwxr-xr-x  - root 15 Feb 22:11 data.db
drwxr-xr-x  - root 15 Feb 22:11 data.db-shm
drwxr-xr-x  - root 15 Feb 22:11 data.db-wal
drwxr-xr-x  - root 15 Feb 22:10 jobs
drwxr-xr-x  - root 15 Feb 22:10 lib
drwxr-xr-x  - root 15 Feb 22:10 host.rb
drwxr-xr-x  - root 15 Feb 22:10 app_restart.rb
drwxr-xr-x  - root 15 Feb 22:10 app_shutdown.rb
drwxr-xr-x  - root 15 Feb 22:10 host_inventory.rb
drwxr-xr-x  - root 15 Feb 22:10 host_report.rb
-rw-r--r-- 408 root 15 Feb 22:10 access.log
-rw-r--r-- 135 root 15 Feb 22:10 error.log
-rw-r--r-- 133 root 15 Feb 22:10 localhost.crt
-rw-r--r-- 165 root 15 Feb 22:10 localhost.key
-rw-r--r-- 164 root 15 Feb 22:10 service.log
-rw-r--r-- 12k root 15 Feb 22:16 .0.0k root 15 Feb 22:14 .0.0k root 15 Feb 22:10 .0.0k root 15 Feb 22:10 .0.0k root 15 Feb 22:15
```

# Criação de Jobs

- Novos jobs devem ser adicionados ao diretório /opt/ecmo/jobs
- Jobs são escritos em Ruby e possuem extensão \*.rb
- Para uso do mitamae, referir-se à [Wiki](#) do projeto
- Deve-se adicionar um cabeçalho ao topo do arquivo contendo os metadados a seguir:

```
# name:           < Nome a ser exibido na interface gráfica >
# schedule:      < “on-demand” ou string no formato crontab >
# log:            < nível de log: debug|info|warn|error >
# revision:      < número indicando a versão do job >
```

# Logs de execução



Host Inventory v1

members:

- avahi

mariane:

name: mariane

gid: 1001

members: []

lightdm:

name: lightdm

gid: 115

members:

- lightdm

sddm:

name: sddm

gid: 116

members: []

mysql:

name: mysql

gid: 117

members:

- mysql

INFO : Recipe: /opt/ecmo/jobs/host\_report.rb

INFO : Recipe: /opt/ecmo/jobs/lib/host.rb

fish /home/gabriel

```
gabriel@avenger ~> doas exa --long --tree --sort=type /opt/ecmo
drwxr-xr-x - root 15 Feb 22:11 /opt/ecmo
drwxr-xr-x - root 15 Feb 22:10 |
  .rwxr-xr-x 20M root 15 Feb 22:10 bin
  .rwxr-xr-x 2.0M root 15 Feb 22:10 ecmo
  .rwxr-xr-x 4.3M root 15 Feb 22:10 mitamae
drwxr-xr-x - root 15 Feb 22:11 db
  .rw-r--r-- 451k root 15 Feb 22:22 data.db
  .rw-r--r-- 33k root 15 Feb 22:24 data.db-shm
  .rw-r--r-- 4.3M root 15 Feb 22:24 data.db-wal
drwxr-xr-x - root 15 Feb 22:10 jobs
drwxr-xr-x - root 15 Feb 22:10 lib
  .rw-r--r-- 408 root 15 Feb 22:10 host.rb
  .rw-r--r-- 135 root 15 Feb 22:10 app_restart.rb
  .rw-r--r-- 133 root 15 Feb 22:10 app_shutdown.rb
  .rw-r--r-- 165 root 15 Feb 22:10 host_inventory.rb
  .rw-r--r-- 164 root 15 Feb 22:10 host_report.rb
  .rw-r--r-- 12k root 15 Feb 22:16 access.log
  .rw----- 3.0k root 15 Feb 22:14 error.log
  .rw----- 1.8k root 15 Feb 22:10 localhost.crt
  .rw----- 3.3k root 15 Feb 22:10 localhost.key
  .rw----- 6.0k root 15 Feb 22:15 service.log
```

# Dashboards

**Informações do Host, Monitoramento de Recursos,  
Armazenamento, Processos**



Logout

Logged in as gabriel/regular

# Dashboard

## General System Metrics

### System Information for avenger

▶ Machine Info

▶ Hardware Resources

▶ Net I/O

### Disk Details

▶ Mountpoint /

https://127.0.0.1:3000/dashboard/

▶ 3000 - dbus-daemon

▶ 3067 - dconf-service

▶ 2389 - earlyoom

▼ 7382 - ecmo

Command

/opt/ecmo/bin/ecmo

Memory

12432 kB

Threads

4

State

R (running)

Parent

7028

▶ 2590 - elogind

▶ 4809 - epiphany

# API de métricas

**Permite o consumo de informações em formato JSON, visando integração com outras soluções**



```
[{"partition_id":1,"id":1,"seconds":1644973985,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":2,"id":2,"seconds":1644973985,"size_mb":10.0,"used_mb":0.0,"free_mb":10.0,"usage":0.0}, {"partition_id":3,"id":3,"seconds":1644973985,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":4,"id":4,"seconds":1644973985,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":5,"id":5,"seconds":1644973985,"size_mb":2931.0859375,"used_mb":0.0,"free_mb":2931.0859375,"usage":0.0}, {"partition_id":6,"id":6,"seconds":1644973985,"size_mb":463820.01953125,"used_mb":273604.41015625,"free_mb":166583.0625,"usage":58.989}, {"partition_id":7,"id":7,"seconds":1644973985,"size_mb":1172.4375,"used_mb":1.1875,"free_mb":1171.25,"usage":0.101}, {"partition_id":8,"id":8,"seconds":1644973985,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":9,"id":9,"seconds":1644973985,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":10,"id":10,"seconds":1644973985,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":11,"id":11,"seconds":1644973985,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":12,"id":12,"seconds":1644973985,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":13,"id":13,"seconds":1644973985,"size_mb":510.98046875,"used_mb":0.2734375,"free_mb":510.70703125,"usage":0.054}, {"partition_id":14,"id":14,"seconds":1644973985,"size_mb":4.0,"used_mb":0.0,"free_mb":4.0,"usage":0.0}, {"partition_id":15,"id":15,"seconds":1644973985,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":16,"id":16,"seconds":1644973985,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":17,"id":17,"seconds":1644973985,"size_mb":586.21484375,"used_mb":0.046875,"free_mb":586.16796875,"usage":0.008}, {"partition_id":18,"id":18,"seconds":1644973985,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":19,"id":19,"seconds":1644973985,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":1,"id":20,"seconds":1644974045,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":2,"id":21,"seconds":1644974045,"size_mb":10.0,"used_mb":0.0,"free_mb":10.0,"usage":0.0}, {"partition_id":3,"id":22,"seconds":1644974045,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":4,"id":23,"seconds":1644974045,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":5,"id":24,"seconds":1644974045,"size_mb":2931.0859375,"used_mb":0.0,"free_mb":2931.0859375,"usage":0.0}, {"partition_id":6,"id":25,"seconds":1644974045,"size_mb":463820.01953125,"used_mb":273604.91015625,"free_mb":166582.5625,"usage":58.989}, {"partition_id":7,"id":26,"seconds":1644974045,"size_mb":1172.4375,"used_mb":1.1875,"free_mb":1171.25,"usage":0.101}, {"partition_id":8,"id":27,"seconds":1644974045,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":9,"id":28,"seconds":1644974045,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":10,"id":29,"seconds":1644974045,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":11,"id":30,"seconds":1644974045,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":12,"id":31,"seconds":1644974045,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":13,"id":32,"seconds":1644974045,"size_mb":510.98046875,"used_mb":0.2734375,"free_mb":510.70703125,"usage":0.054}, {"partition_id":14,"id":33,"seconds":1644974045,"size_mb":4.0,"used_mb":0.0,"free_mb":4.0,"usage":0.0}, {"partition_id":15,"id":34,"seconds":1644974045,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":16,"id":35,"seconds":1644974045,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":17,"id":36,"seconds":1644974045,"size_mb":586.21484375,"used_mb":0.046875,"free_mb":586.16796875,"usage":0.008}, {"partition_id":18,"id":37,"seconds":1644974045,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":19,"id":38,"seconds":1644974045,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":1,"id":39,"seconds":1644974104,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":2,"id":40,"seconds":1644974104,"size_mb":10.0,"used_mb":0.0,"free_mb":10.0,"usage":0.0}, {"partition_id":3,"id":41,"seconds":1644974104,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":4,"id":42,"seconds":1644974104,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}, {"partition_id":5,"id":43,"seconds":1644974104,"size_mb":2931.0859375,"used_mb":0.0,"free_mb":2931.0859375,"usage":0.0}, {"partition_id":6,"id":44,"seconds":1644974104,"size_mb":463820.01953125,"used_mb":273605.79296875,"free_mb":166581.6796875,"usage":58.99}, {"partition_id":7,"id":45,"seconds":1644974104,"size_mb":1172.4375,"used_mb":1.1875,"free_mb":1171.25,"usage":0.101}, {"partition_id":8,"id":46,"seconds":1644974104,"size_mb":0.0,"used_mb":0.0,"free_mb":0.0,"usage":0.0}
```

127.0.0.1:3000/metrics/mem X



https://127.0.0.1:3000/metrics/memory



JSON Raw Data Headers

Save Copy Collapse All Expand All (slow) Filter JSON

▼ 0:

```
id:      1
seconds: 1644973932
total_mb: 5862.171875
used_mb:  1775.578125
free_mb:  4086.59375
```

▼ 1:

```
id:      2
seconds: 1644973992
total_mb: 5862.171875
used_mb:  1786.34765625
free_mb:  4075.82421875
```

▼ 2:

```
id:      3
seconds: 1644974052
total_mb: 5862.171875
used_mb:  1792.7265625
free_mb:  4069.4453125
```

▼ 3:

```
id:      4
seconds: 1644974112
total_mb: 5862.171875
used_mb:  1821.1953125
free_mb:  4040.9765625
```

► 4: {...}

► 5: {...}

► 6: {...}

► 7: {...}

► 8: {...}

► 9: {...}

# Desenvolvimento

Overview do stack tecnológico

Home

Forum

Blog

Sponsors

Community

Conference

Team

Docs

GitHub



# CRYSTAL

A language for humans and computers

INSTALL

LEARN

TRY ONLINE

Fund Crystal and help it keep growing at [open collective](#)

Latest release 1.3.2

## SYNTAX

Crystal's syntax is heavily inspired by Ruby's, so it feels natural to read and easy to write, and has the added benefit of a lower learning curve for experienced Ruby devs.

```
# A very basic HTTP server
require "http/server"

server = HTTP::Server.new do |context|
    context.response.content_type = "text/plain"
    context.response.print "Hello world, got #{context.request.path}!"
end

puts "Listening on http://127.0.0.1:8080"
server.listen(8080)
```

[Start learning Crystal with the Language Reference](#)



# Spider-Gazelle

Developer Guide

[GitHub](#)[Get Started](#)

## Actix:

```
App::new()
    .resource("/", |r| r.f(|r| HttpResponse::build_from(r)))
    .resource("/user", |r| r.f(|r| HttpResponse::build_from(r)))
    .resource("/user/{id}", |r| {
        r.f(|req| {
            let id = Binary::from_slice(req.match_info()["id"].as_ref());
            HttpResponse::build_from(req).body(id)
        })
    })
}
```

## Spider-Gazelle:

```
class User < Application
    def index
        head :ok
    end
    def show
        render text: route_params["id"]
    end
    def create
        head :ok
    end
end
```

## Rails:

```
class ApiController < ApplicationController
    def index
        head 200
    end
    def user
        render plain: params['id']
    end
    def register_user
        head 200
    end
end
```



## Spider-Gazelle Guide

## Getting started

[Overview](#)[Creating a new project](#)[Deploying your project](#)

## Models

[Model basics](#)[Validations](#)

## Controllers

[Overview](#)[Routing](#)[WebSockets](#)

## Views

[Templates](#)

## Spider-Gazelle

An elegant web framework designed to be simple, powerful and fast.

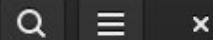
Spider-Gazelle is one of the fastest web frameworks. Without compromising developer productivity / happiness.

▼ Ranked by requests (ordered by number of requests per second - highest is better)

Language (Runtime)	Framework (Middleware)	Requests / s	Throughput
rust	actix-web	190682.67	216.81 MB
cpp	evhttp	163378.00	158.60 MB
python	vibora	148099.00	168.17 MB
go	fasthttprouter	146274.00	235.79 MB
crystal	spider-gazelle	140155.00	149.91 MB
java	act	127374.00	217.87 MB
python	tornado	8603.33	22.76 MB
ruby	rails	4705.33	14.33 MB



## glances /home/gabriel/Public



avenger (Alpine Linux 3.15.0 64bit / Linux 5.15.23-0-lts)

Uptime: 0:23:01

CPU -										MEM -										LOAD	
idle:		ctx_sw:		2K:		total:		inactive:		389M:		used:		buffers:		free:		cached:		total:	
user:		irq:		inter:		6K:		nice:		1.39G:		1.41G:		116M:		0:		1.04G:		0:	
system:		sw_int:		709:		used:		1.41G:		1.39G:		1.41G:		116M:		0:		1.04G:		0:	
iowait:		steal:		0.9%:		0.0%:		0.0%:		0.0%:		4.31G:		1.04G:		4.00G:		4.00G:		0.98:	
4-core																					

## NETWORK

	Rx/s	Tx/s	TASKS	204	(499 thr), 1 run, 152 slp, 51 oth sorted automatically by CPU consumption
lo	0b	0b			
wlan0	0b	0b	CPU%	MEM%	VIRT RES PID USER TIME+ THR NI S R/s W/s Command ('k' to kill)

## DISK I/O

	R/s	W/s	CPU%	MEM%	VIRT	RES	PID	USER	TIME+	THR	NI	S	R/s	W/s	Command ('k' to kill)
sda	0	17K	0.7	0.0	1.55M	52K	2435	root	0:02	1	0	R	0	0	python3 /usr/bin/glances
sda1	0	0	0.5	5.6	12.4G	331M	4133	gabriel	1:05	15	0	S	0	0	gnome-shell
sda2	0	0	0.5	0.8	111M	48.4M	4404	gabriel	0:03	1	0	S	0	0	acpid
sda3	0	17K	0.5	0.0	0	0	1768	root	0:02	1	0	D	?	?	editors_helper --type=zygote --no-sandbox --force-device-s
sr0	0	0	0.2	3.2	2.32G	187M	4006	gabriel	0:12	24	0	S	0	0	gnome-terminal-server
			0.2	0.2	142M	12.7M	7382	root	0:00	4	0	S	?	?	[irq/39-ELAN0501]
															DesktopEditors

## FILE SYS

	Used	Total	0.0	1.8	166M	107M	3344	gabriel	0:08	4	0	S	0	0	gnome-software --gapplication-service
/ (sda3)	267G	453G	0.0	1.7	1.16G	102M	4117	gabriel	0:00	9	0	S	0	0	editors_helper --type=zygote --no-sandbox --force-device-s

## SENSORS

Package id 0	48C	0.0	1.1	91.2M	66.2M	3348	gabriel	0:01	4	0	S	0	0	0	gnome-clocks -b
Core 0	46C	0.0	1.1	123M	63.2M	6390	gabriel	0:05	5	0	S	0	0	0	nautilus --gapplication-service
Core 1	47C	0.0	1.1	789M	63.1M	4107	gabriel	0:00	7	0	S	0	0	0	editors_helper --type=utility --field-trial-handle=2938635
BAT BAT1	100%	0.0	1.1	2.13G	62.6M	6439	gabriel	0:00	8	0	S	0	0	0	gjs /home/gabriel/.local/share/gnome-shell/extensions/ddte
		0.0	0.9	285M	51.5M	4072	gabriel	0:00	1	0	S	0	0	0	editors_helper --type=zygote --no-sandbox --force-device-s
		0.0	0.7	151M	40.1M	3060	gabriel	0:00	4	0	S	0	0	0	goa-daemon
		0.0	0.5	75.6M	31.9M	3351	gabriel	0:03	6	19	S	0	0	0	tracker-miner-fs-3
		0.0	0.5	87.7M	30.6M	3079	gabriel	0:00	9	0	S	0	0	0	evolution-calendar-factory
		0.0	0.5	51.8M	28.3M	3012	gabriel	0:06	2	0	S	0	0	0	Xwayland :0 -rootless -noreset -accessx -core -auth /run/u
		0.0	0.5	54.7M	27.3M	3799	gabriel	0:00	4	0	S	0	0	0	seahorse --gapplication-service
		0.0	0.5	68.0M	27.1M	3092	gabriel	0:00	6	0	S	0	0	0	evolution-addressbook-factory
		0.0	0.5	61.5M	26.8M	3165	gabriel	0:00	5	0	S	0	0	0	gsd-xsettings

**Obrigado por utilizar o Ecmo!**

## Ecmo: Easy Configuration Management Orchestration

Your session is finished. Thank you for using Ecmo.

 Username PasswordLogin