

Zammad

Zammad

- [Instalar Zammad](#)
- [Cambiar contraseña zammad](#)
- [Borrar ticket en Zammad](#)
- [Cambiar estado ticket zammad](#)

Instalar Zammad

Instalación en debian 12

Preparar el entorno

```
apt update
apt install curl apt-transport-https gnupg
```

Instalar ELASTICSEARCH

```
apt install apt-transport-https sudo wget curl gnupg
echo "deb [signed-by=/etc/apt/trusted.gpg.d/elasticsearch.gpg]
https://artifacts.elastic.co/packages/7.x/apt stable main" | \
tee -a /etc/apt/sources.list.d/elastic-7.x.list > /dev/null
curl -fsSL https://artifacts.elastic.co/GPG-KEY-elasticsearch | \
gpg --dearmor | tee /etc/apt/trusted.gpg.d/elasticsearch.gpg> /dev/null
apt update
apt install elasticsearch
/usr/share/elasticsearch/bin/elasticsearch-plugin install ingest-attachment
```

Configurar elasticsearch para inicio automático

```
systemctl start elasticsearch
systemctl enable elasticsearch
```

Añadimos a /etc/elasticsearch/elasticsearch.yml

```
http.max_content_length: 400mb

indices.query.bool.max_clause_count: 2000
```

reiniciamos servicio elasticsearch

```
systemctl restart elasticsearch
```

Habilitamos el repositorio de Zammad

```
curl -fsSL https://dl.packager.io/srv/zammad/zammad/key | \
  gpg --dearmor | tee /etc/apt/trusted.gpg.d/pkgr-zammad.gpg> /dev/null

echo "deb [signed-by=/etc/apt/trusted.gpg.d/pkgr-zammad.gpg]
https://dl.packager.io/srv/deb/zammad/zammad/stable/debian 12 main"| \
  tee /etc/apt/sources.list.d/zammad.list > /dev/null
```

Instalar Zammad

```
apt update
apt install zammad
```

Conectar Zammad a elasticsearch

```
zammad run rails r "Setting.set('es_url', 'http://localhost:9200')"
zammad run rake zammad:searchindex:rebuild
```

Opcionalmente podemos poner el rebuild con el número de hilos (cores) para mejorar el tiempo de respuesta. En el ejemplo 8 cores

```
zammad run rake zammad:searchindex:rebuild[8]
```

Copiamos el fichero de configuración de nginx de ejemplo

Configurar Nginx con certificado

```
cp /opt/zammad/contrib/nginx/zammad.conf /etc/nginx/sites-enabled/zammad.conf
```

A continuación instalaremos el certbot para poner el certificado.

Instalación para debian 10

Preparar el entorno

```
$ apt update
$ apt install curl

# For Debian 10+
$ apt install nodejs
```

Instalar ELASTICSEARCH

```
$ apt install apt-transport-https sudo wget
$ echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | tee -a /etc/apt/sources
$ wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | apt-key add -
$ apt update
$ apt install elasticsearch
$ /usr/share/elasticsearch/bin/elasticsearch-plugin install ingest-attachment
responder S

$ systemctl start elasticsearch
$ systemctl enable elasticsearch
```

Añadimos a /etc/elasticsearch/elasticsearch.yml

```
http.max_content_length: 400mb

indices.query.bool.max_clause_count: 2000
```

reiniciamos servicio elasticsearch

```
$ systemctl restart elasticsearch
```

Continuamos instalando zammad

locale |grep "LANG="

debe devolver <lang_code>.utf8

Agregar repositorio Zammad

```
wget -qO- https://dl.packager.io/srv/zammad/zammad/key | apt-key add -
```

```
wget -O /etc/apt/sources.list.d/zammad.list \
https://dl.packager.io/srv/zammad/zammad/stable/installer/debian/10.repo
```

Instalamos zammad

```
$apt update
$apt install zammad
```

Otros aspectos a tener en cuenta

OJO AL FIREWALL DEBIAN

IGUAL HAY QUE AÑADIR

```
#!/usr/local/sbin/nft -f
flush ruleset

table inet filter {
    chain input {
        type filter hook input priority 0; policy drop;
        ct state established,related accept
        tcp dport ssh log accept
        tcp dport { http, https } accept
        udp dport { http, https } accept
    }

    chain forward {
        type filter hook forward priority 0; policy accept;
    }

    chain output {
        type filter hook output priority 0; policy accept;
    }
}
```

Conecta Zammad

```
$zammad run rails r "Setting.set('es_url', 'http://localhost:9200')"
$zammad run rake searchindex:rebuild
```

Configurar Nginx con certificado

```
cp /opt/zammad/contrib/nginx/zammad_ssl.conf /etc/nginx/sites-enabled/zammad.conf
```

poner los certificados

y configurar via web la organizacion

Cambiar contraseña zammad

Se abre consola Rails:

```
zammad run rails c
```

Se lanza la siguiente sentencia:

```
User.find_by(email: 'usuario@dominio.com').update!(password: 'micontraseña')
```

Borrar ticket en Zammad

Muchas veces nos encontramos con un ticket de spam, que lo cerramos, pero al enviar respuesta, vuelve a abrirse, una y otra vez.

Para eliminarlo entramos en la consola de rails

Ejecutamos

```
zammad run rails c
```

Y una vez nos aparezca la consola (tardará unos 30/40 segundos)

Tecleamos Ticket.find(XYZ) Y comprobamos que es realmente el ticket que queremos eliminar (en nuestro caso es el 348)

```
=> #<Ticket id: 348, group_id: 1, priority_id: 2, state_id: 2, organization_id: nil, number:
"35003463",
title: "Spam subject", owner_id: 3, customer_id: 114, note: nil, first_response_at: "2022-02-
28 10: 24: 43",
first_response_escalation_at: nil, first_response_in_min: 490027, first_response_diff_in_min:
-490012,
close_at: "2021-12-27 13: 28: 22", close_escalation_at: nil, close_in_min: 490027,
close_diff_in_min: -489667, update_escalation_at: nil, update_in_min: nil,
update_diff_in_min: nil, last_contact_at: "2022-03-05 11: 20: 23",
last_contact_agent_at: "2022-02-28 10: 24: 43", last_contact_customer_at: "2022-03-05 11: 20: 23",

last_owner_update_at: "2022-04-16 08: 01: 57", create_article_type_id: 1,
create_article_sender_id: 2,
article_count: 21, escalation_at: nil, pending_time: nil, type: nil, time_unit: nil,
preferences: {"channel_id"=>3, "escalation_calculation"=>{"first_response_at"=>Mon, 28 Feb
2022 10: 24: 43 UTC +00: 00,
  "last_update_at"=>Sat, 05 Mar 2022 11: 20: 23 UTC +00: 00, "close_at"=>Mon, 27 Dec 2021
13: 28: 22 UTC +00: 00,
  "last_contact_at"=>Sat, 05 Mar 2022 11: 20: 23 UTC +00: 00, "sla_id"=>2, "sla_updated_at"=>Mon,
13 Apr 2020 20: 33: 25 UTC +00: 00,
  "calendar_id"=>2, "calendar_updated_at"=>Fri, 15 Apr 2022 14: 00: 22 UTC +00: 00,
  "escalation_disabled"=>false}},
```



```
updated_by_id: 117, created_by_id: 114, created_at: "2021-01-21 06:20:56", updated_at: "2022-04-16 08:01:59"
```

Y para eliminarlo

```
irb(main):003:0>Ticket.find( XYZ ).destroy
```

Donde XYZ es el id del ticket que se ve en la URL

<https://soporte.ateinco.net/#ticket/zoom/XYZ>

Cambiar estado ticket zammad

Para ello deberemos de entrar en la consola de Zammad

```
zammad run rails c
```

Y una vez nos aparezca la consola (tardará unos 30/40 segundos)

Tecleamos `Ticket.find(XYZ)` Y comprobamos que es realmente el ticket al que queremos cambiar es estado

```
=> #<Ticket id: 348, group_id: 1, priority_id: 2, state_id: 2, organization_id: nil, number:
"35003463",
title: "Spam subject", owner_id: 3, customer_id: 114, note: nil, first_response_at: "2022-02-
28 10: 24: 43",
first_response_escalation_at: nil, first_response_in_min: 490027, first_response_diff_in_min:
-490012,
close_at: "2021-12-27 13: 28: 22", close_escalation_at: nil, close_in_min: 490027,
close_diff_in_min: -489667, update_escalation_at: nil, update_in_min: nil,
update_diff_in_min: nil, last_contact_at: "2022-03-05 11: 20: 23",
last_contact_agent_at: "2022-02-28 10: 24: 43", last_contact_customer_at: "2022-03-05 11: 20: 23",

last_owner_update_at: "2022-04-16 08: 01: 57", create_article_type_id: 1,
create_article_sender_id: 2,
article_count: 21, escalation_at: nil, pending_time: nil, type: nil, time_unit: nil,
preferences: {"channel_id"=>3, "escalation_calculation"=>{"first_response_at"=>Mon, 28 Feb
2022 10: 24: 43 UTC +00: 00,
  "last_update_at"=>Sat, 05 Mar 2022 11: 20: 23 UTC +00: 00, "close_at"=>Mon, 27 Dec 2021
13: 28: 22 UTC +00: 00,
  "last_contact_at"=>Sat, 05 Mar 2022 11: 20: 23 UTC +00: 00, "sla_id"=>2, "sla_updated_at"=>Mon,
13 Apr 2020 20: 33: 25 UTC +00: 00,
  "calendar_id"=>2, "calendar_updated_at"=>Fri, 15 Apr 2022 14: 00: 22 UTC +00: 00,
"escalation_disabled"=>false}},
updated_by_id: 117, created_by_id: 114, created_at: "2021-01-21 06: 20: 56", updated_at: "2022-
04-16 08: 01: 59"
```

Una vez comprobado, cambiamos el estado

```
Ticket.find(348).update(state_id: 4)
```