# ABOUT

1 Zammad

2 Software
   2.1 Client requirements
   2.2 Server requirements

3 Hardware
   3.1 For Zammad and a database server like PostgreSQL we recommend at least:
   3.2 For optimal performance up to 40 agents:
   3.3 Performance Tuning

4 Install from package
   4.1 Prerequisites
   4.2 Add Repository and install Zammad
   4.3 Firewall & SELinux
   4.4 Manage services of Zammad
   4.5 Next steps

5 Install from source
   5.1 Install from source on MacOS
   5.2 Prerequisites
   5.3 Installation
   5.4 Manage services of Zammad
   5.5 Firewall & SELinux
   5.6 Next steps

6 Set up Elasticsearch
   6.1 Step 1: Installation
   6.2 Step 2: Suggested Configuration
   6.3 Step 3: Connect Zammad
   6.4 Appendix

7 Install with Docker Compose
   7.1 Install Docker Environment
   7.2 Getting started with zammad-docker-compose
   7.3 Next steps

8 Install on Kubernetes via Helm

9 Installation on Univention Corporate Server via App Center (EOL)
   9.1 Prerequisites
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.2 Installing Zammad</td>
<td>60</td>
</tr>
<tr>
<td>9.3 First steps you should consider</td>
<td>60</td>
</tr>
<tr>
<td>9.4 Further information</td>
<td>61</td>
</tr>
<tr>
<td>10 Updating Zammad</td>
<td>69</td>
</tr>
<tr>
<td>10.1 Updating Elasticsearch</td>
<td>72</td>
</tr>
<tr>
<td>11 Configure the webserver</td>
<td>73</td>
</tr>
<tr>
<td>11.1 Get a ssl certificate (recommended)</td>
<td>73</td>
</tr>
<tr>
<td>11.2 Adjusting the webserver configuration</td>
<td>75</td>
</tr>
<tr>
<td>12 First steps</td>
<td>81</td>
</tr>
<tr>
<td>12.1 Getting Started Wizard</td>
<td>81</td>
</tr>
<tr>
<td>12.2 Further Steps</td>
<td>82</td>
</tr>
<tr>
<td>13 Migrating to Zammad</td>
<td>85</td>
</tr>
<tr>
<td>13.1 Limitations</td>
<td>85</td>
</tr>
<tr>
<td>13.2 Available Migration Options</td>
<td>85</td>
</tr>
<tr>
<td>14 Console</td>
<td>105</td>
</tr>
<tr>
<td>14.1 Start Zammad's Rails console</td>
<td>105</td>
</tr>
<tr>
<td>14.2 Working on the console</td>
<td>106</td>
</tr>
<tr>
<td>15 Start</td>
<td>123</td>
</tr>
<tr>
<td>15.1 Source Code</td>
<td>123</td>
</tr>
<tr>
<td>15.2 Documentation</td>
<td>123</td>
</tr>
<tr>
<td>16 Supported Branches / Versions</td>
<td>125</td>
</tr>
<tr>
<td>16.1 develop</td>
<td>125</td>
</tr>
<tr>
<td>16.2 stable</td>
<td>125</td>
</tr>
<tr>
<td>16.3 stable-x.y</td>
<td>125</td>
</tr>
<tr>
<td>17 Introduction</td>
<td>127</td>
</tr>
<tr>
<td>17.1 API clients</td>
<td>127</td>
</tr>
<tr>
<td>17.2 Authentication</td>
<td>127</td>
</tr>
<tr>
<td>17.3 Endpoints and example data</td>
<td>128</td>
</tr>
<tr>
<td>17.4 Content Type</td>
<td>128</td>
</tr>
<tr>
<td>17.5 Response Payloads (expand)</td>
<td>128</td>
</tr>
<tr>
<td>17.6 Pagination</td>
<td>137</td>
</tr>
<tr>
<td>17.7 Sorting search results</td>
<td>137</td>
</tr>
<tr>
<td>17.8 Actions on behalf of other users</td>
<td>137</td>
</tr>
<tr>
<td>17.9 Encoding</td>
<td>138</td>
</tr>
<tr>
<td>18 User</td>
<td>139</td>
</tr>
<tr>
<td>18.1 me - current user</td>
<td>139</td>
</tr>
<tr>
<td>18.2 List</td>
<td>142</td>
</tr>
<tr>
<td>18.3 Search</td>
<td>157</td>
</tr>
<tr>
<td>18.4 Show</td>
<td>160</td>
</tr>
<tr>
<td>18.5 Create</td>
<td>161</td>
</tr>
<tr>
<td>18.6 Update</td>
<td>164</td>
</tr>
<tr>
<td>18.7 Delete</td>
<td>166</td>
</tr>
<tr>
<td>19 Organization</td>
<td>167</td>
</tr>
<tr>
<td>19.1 List</td>
<td>167</td>
</tr>
<tr>
<td>19.2 Search</td>
<td>169</td>
</tr>
<tr>
<td>19.3 Show</td>
<td>170</td>
</tr>
<tr>
<td>Chapter</td>
<td>Section</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>19.4</td>
<td>Create</td>
</tr>
<tr>
<td>19.5</td>
<td>Update</td>
</tr>
<tr>
<td>19.6</td>
<td>Delete</td>
</tr>
<tr>
<td>20</td>
<td>Group</td>
</tr>
<tr>
<td>20.1</td>
<td>List</td>
</tr>
<tr>
<td>20.2</td>
<td>Show</td>
</tr>
<tr>
<td>20.3</td>
<td>Create</td>
</tr>
<tr>
<td>20.4</td>
<td>Update</td>
</tr>
<tr>
<td>20.5</td>
<td>Delete</td>
</tr>
<tr>
<td>21</td>
<td>Calendar</td>
</tr>
<tr>
<td>21.1</td>
<td>List</td>
</tr>
<tr>
<td>21.2</td>
<td>Show</td>
</tr>
<tr>
<td>21.3</td>
<td>Create</td>
</tr>
<tr>
<td>21.4</td>
<td>Update</td>
</tr>
<tr>
<td>21.5</td>
<td>Delete</td>
</tr>
<tr>
<td>22</td>
<td>Service-Level Agreements (SLA)</td>
</tr>
<tr>
<td>22.1</td>
<td>List</td>
</tr>
<tr>
<td>22.2</td>
<td>Show</td>
</tr>
<tr>
<td>22.3</td>
<td>Create</td>
</tr>
<tr>
<td>22.4</td>
<td>Update</td>
</tr>
<tr>
<td>22.5</td>
<td>Delete</td>
</tr>
<tr>
<td>23</td>
<td>Ticket endpoints</td>
</tr>
<tr>
<td>23.1</td>
<td>Tickets</td>
</tr>
<tr>
<td>23.2</td>
<td>Articles</td>
</tr>
<tr>
<td>23.3</td>
<td>Priorities</td>
</tr>
<tr>
<td>23.4</td>
<td>States</td>
</tr>
<tr>
<td>23.5</td>
<td>Tags</td>
</tr>
<tr>
<td>23.6</td>
<td>Linking Tickets</td>
</tr>
<tr>
<td>23.7</td>
<td>Mentions</td>
</tr>
<tr>
<td>24</td>
<td>Generic CTI</td>
</tr>
<tr>
<td>24.1</td>
<td>New call</td>
</tr>
<tr>
<td>24.2</td>
<td>Call hangup</td>
</tr>
<tr>
<td>24.3</td>
<td>Call answered</td>
</tr>
<tr>
<td>25</td>
<td>Online Notification</td>
</tr>
<tr>
<td>25.1</td>
<td>List</td>
</tr>
<tr>
<td>25.2</td>
<td>Show</td>
</tr>
<tr>
<td>25.3</td>
<td>Update</td>
</tr>
<tr>
<td>25.4</td>
<td>Delete</td>
</tr>
<tr>
<td>25.5</td>
<td>Mark all as read</td>
</tr>
<tr>
<td>26</td>
<td>Object</td>
</tr>
<tr>
<td>26.1</td>
<td>List</td>
</tr>
<tr>
<td>26.2</td>
<td>Show</td>
</tr>
<tr>
<td>26.3</td>
<td>Create</td>
</tr>
<tr>
<td>26.4</td>
<td>Update</td>
</tr>
<tr>
<td>26.5</td>
<td>Delete</td>
</tr>
<tr>
<td>26.6</td>
<td>Execute Database Migrations</td>
</tr>
<tr>
<td>27</td>
<td>User Access Token</td>
</tr>
</tbody>
</table>
27.1 List ................................................................. 305
27.2 Create .............................................................. 322
27.3 Delete ............................................................... 323

28 Backup and Restore ............................................ 325
28.1 Getting Started .................................................... 325
28.2 Limitations ......................................................... 335

29 Configuration via Environment Variables .................. 337
29.1 General Options ................................................. 337
29.2 Network Options ................................................. 338
29.3 Performance Tuning ............................................. 338

30 Configure Database server .................................... 341

31 Privacy & Data Retention ..................................... 343
31.1 On-Premises Data .............................................. 343
31.2 External Services ............................................... 344

32 Troubleshooting and FAQ .................................... 345
32.1 Data missing from the Web-UI / Search data missing or incomplete .............................................. 345

33 Single Sign-On for Kerberos ................................. 349
33.1 Conceptual Overview ......................................... 349
33.2 Getting Started .................................................. 350
33.3 Step 1: Configure Active Directory ...................... 350
33.4 Step 2: Remove NGINX, Set up Apache + Kerberos .................................................. 353
33.5 Step 3: Enable SSO in Zammad ......................... 356
33.6 Step 4: Configure Client System (Windows Only) ...... 357
33.7 Troubleshooting ............................................... 358

34 Reporting Tools (Third party) ............................... 361
34.1 Getting Started .................................................. 361
34.2 Third Party Reporting Tools known to be working .............................................. 362
The Zammad documentation consists of three parts:

- Zammad system installation and configuration (this documentation)
- Zammad administration (https://admin-docs.zammad.org)
- Zammad user documentation (https://user-docs.zammad.org)
Do you receive many emails and want to answer them with a team of agents?

You're going to love Zammad!

Zammad is a web based open source helpdesk/customer support system with many features to manage customer communication via several channels like telephone, facebook, twitter, chat and emails. It is distributed under version 3 of the GNU AFFERO General Public License (GNU AGPLv3).

The code is open source, and available on GitHub!
2.1 1. Client requirements

Please note that, while Zammad being a web application, there’s some requirements for your clients. This ensures that Zammad works as expected.

### 2.1.1 1.1. Supported Browsers

Table 1: Zammad/Browser version compatibility

<table>
<thead>
<tr>
<th>Browser</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Firefox 78+</td>
<td>(ESR)</td>
</tr>
<tr>
<td>(Google) Chrome 83+</td>
<td>This also applies for all Chromium based Browsers like Microsoft Edge</td>
</tr>
<tr>
<td>Opera 69+</td>
<td>(based on Chromium 83)</td>
</tr>
<tr>
<td>Microsoft Internet Explorer 11</td>
<td></td>
</tr>
<tr>
<td>Safari 11</td>
<td></td>
</tr>
</tbody>
</table>

**Note:**

Zammad heavily uses Javascript which makes it a hard requirement.

Browser addons that hook into page content may interfere with Zammad's function which is not a bug. Google Chromes translation module is known to do odd things to especially state names. Use Zammad's internal translations instead.

### 2.1.2 1.2. Network requirements

Zammad uses web sockets. Some application firewalls may filter these connections. This may lead to decreased browser performance.

There's a fallback to Ajax which causes a higher application server load and thus should be avoided.

**Note:** The “Core workflows” feature of Zammad comes with an Ajax Mode which can be activated separately in case you're having issues with field selections.
2.2 2. Server requirements

If you want to install Zammad, you need the following software.

Note: Most of the software versions listed below (unless stated as specific version) are minimum requirements of Zammad. We strongly encourage you to use most current possible versions that are not end of life.

2.2.1 2.1. Ruby Programming Language

Hint: Only relevant for source code installations

Docker and package installations provide the required ruby dependencies!

Zammad requires Ruby. All required rubygems like ruby on rails are listed in the Gemfile.

The following Ruby version is supported:

Ruby 3.0.4

<table>
<thead>
<tr>
<th>Zammad</th>
<th>Ruby</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.2+</td>
<td>3.0.4</td>
</tr>
<tr>
<td>5.0 - 5.1</td>
<td>2.7.4</td>
</tr>
<tr>
<td>3.4.1 - 4.1</td>
<td>2.6.6</td>
</tr>
<tr>
<td>3.4.0</td>
<td>2.6.5</td>
</tr>
<tr>
<td>3.1 - 3.3</td>
<td>2.5.5</td>
</tr>
<tr>
<td>2.5 - 3.0</td>
<td>2.4.4</td>
</tr>
</tbody>
</table>

2.2.2 2.2. Supported distributions

Below you can find all distributions Zammad provides packages for.

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Versions</th>
</tr>
</thead>
<tbody>
<tr>
<td>CentOS / RHEL</td>
<td>7 &amp; 8</td>
</tr>
<tr>
<td>Debian</td>
<td>9, 10 &amp; 11</td>
</tr>
<tr>
<td>OpenSuSE / SLES</td>
<td>Leap 42.3 / 12</td>
</tr>
<tr>
<td>Ubuntu</td>
<td>16.04, 18.04 &amp; 20.04</td>
</tr>
</tbody>
</table>

Warning: SuSE users be aware

Due to the age of SLES12 / Leap 42.3 you may no longer be able to satisfy all (soft) dependencies of Zammad.

If you’re not running a docker-compose or package installation please consider changing to a different distribution that’s supported.
Danger: Incompatibility warning for Ubuntu 22.04 LTS

Please note that due to Ubuntu’s dependencies, Zammad currently is not compatible to Ubuntu 22. This affects all installation types except for docker style installations!

Note: What about my specific distribution?! It’s so cool!

If your distribution is not listed, you can still install Zammad. For this you can either use Docker-Compose or Source installation.

We try to provide all current distributions that are supported by Packager.io. This means that we can’t always provide support for your favorite system.

2.2.3 2.3. Package Dependencies

The below dependencies need to be installed on your system. If you’re using the package install, the packages below will automatically installed with the Zammad-Package.

# Debian 9, 10 & 11, Ubuntu 16.04, 18.04 & 20.04
$ apt install libimlib2

# openSUSE
$ zypper install imlib2

# CentOS 7 & 8
$ yum install epel-release
$ yum install imlib2

Note:

libimlib2-dev or imlib2-devel are no longer required.

However: If you have to use bundle install for e.g. custom Gems or development, you’ll need to install it!

2.2.4 2.4. Database Server

Zammad will store all content in a Database. You can choose between the following database servers:

• MySQL 5.7+
• MariaDB 10.3+
• PostgreSQL 9.3+

Note: We tend to recommend PostgreSQL. For the last 10 years we had the best experience with it.

Zammad requires UTF-8 for its database.
Warning: Required configuration for MySQL/MariaDB:

- Use UTF-8 encoding - utf8mb4 for example will fail!
- Set max_allowed_packet to a value larger than the default of 4 MB (64 MB+ recommended).

You may also want to consider the following settings for your MySQL server:

```
innodb_file_format = Barracuda
innodb_file_per_table = on
innodb_default_row_format = dynamic
innodb_large_prefix = 1
innodb_file_format_max = Barracuda
```

2.2.5 2.5. Node.js

Note:

This soft dependency was introduced with Zammad 5.0. Package installations come pre-bundled with the correct NodeJS version. Unless you require NodeJS on your machine for other projects, a manual installation is not required.

Node.js is required for asset compiling.

Hint: No changes to JS or CSS files?

If you don’t require to change any javascript or stylesheed files you’ll be fine without this package. It’s only required if you have to run `rake assets:precompile` on your system.

Warning: Node.js is required on source code installations.

<table>
<thead>
<tr>
<th>Zammad</th>
<th>Node.js</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.2+</td>
<td>16.0+</td>
</tr>
<tr>
<td>5.0 - 5.1</td>
<td>10.0+</td>
</tr>
</tbody>
</table>

2.2.6 2.6. Reverse Proxy

In a typical web environment today, you use a reverse proxy to deliver the static content of your application. Only the “expensive” app required HTTP requests are forwarded to the application server.

The following reverse proxies are supported:

- Nginx 1.3+
- Apache 2.2+
2.2.7 2.7. Elasticsearch (optional)

Zammad uses Elasticsearch to

1) make search faster
2) support advanced features like reports
3) searching by email attachment contents

This becomes increasingly important as the number of tickets in your system gets larger and larger.

This dependency is optional but strongly recommended; Zammad will work without it, but search performance will be degraded, and some features will be disabled.

**Hint:** If you install Zammad via package manager...

It’s perfectly safe to manually override the Elasticsearch dependency. The appropriate command line flag will depend on your platform (e.g., `--force`, `--ignore-depends`, `--skip-broken`); check your package manager’s manpage to find out.

**Warning:** Please note that if you do not install and use Elasticsearch, the search will be very limited! We recommend using Elasticsearch, as it will boost the usage of Zammad greatly!

**Note:** Starting with Zammad 4.0 you can decide if you want to use `elasticsearch` or `elasticsearch-oss`. Please note that CentOS requires `elasticsearch`.

### Table 5: Zammad/Elasticsearch version compatibility

<table>
<thead>
<tr>
<th>Zammad</th>
<th>Elasticsearch</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.2+</td>
<td>&gt;= 7.8, &lt; 9</td>
</tr>
<tr>
<td>5.0 - 5.1</td>
<td>&gt;= 7.8, &lt; 8</td>
</tr>
<tr>
<td>4.0-4.1</td>
<td>&gt;= 6.5, &lt;= 7.12</td>
</tr>
<tr>
<td>3.4-3.6</td>
<td>&gt;= 5.5, &lt;= 7.9</td>
</tr>
<tr>
<td>3.3</td>
<td>&gt;= 2.4, &lt;= 7.6</td>
</tr>
<tr>
<td>3.2</td>
<td>&gt;= 2.4, &lt;= 7.5</td>
</tr>
<tr>
<td>3.1</td>
<td>&gt;= 2.4, &lt;= 7.4</td>
</tr>
<tr>
<td>2.0–3.0</td>
<td>&gt;= 2.4, &lt;= 5.6</td>
</tr>
</tbody>
</table>

An Elasticsearch plugin is required to index the contents of email attachments: `ingest-attachment`.

2.2.8 2.8. Optional tools of improved caching and distribution

**Note:** The features / integrations below were introduced by Zammad 5.0

These tools are optional and may make a lot of sense in big environments even if you decide against distributed use cases.

We consider this topic as *Performance Tuning*. 
2.8.1 Redis

Using Redis allows you to store all web socket information in Redis instead of your file system.

Note: Configuration and installation is out of our scope. Please follow the official vendor guides and ensure to have a tight security on your installation.

2.8.2 Memcached

Instead of storing Zammads cache files within your filesystem, you can also do so in Memcached. This can allow you to restrict the size of your cache directories to improve performance.

Note: Configuration and installation is out of our scope. Please follow the official vendor guides and ensure to have a tight security on your installation.
CHAPTER THREE

HARDWARE

You can run Zammad on bare metal or on a virtual machine. Choose what you prefer.

3.1 For Zammad and a database server like PostgreSQL we recommend at least:

- 2 CPU cores
- 4 GB of RAM (+4 GB if you want to run Elasticsearch on the same server)

3.2 For optimal performance up to 40 agents:

- 4 CPU cores
- 6 GB of RAM (+6 GB if you want to run Elasticsearch on the same server)

Of course at the end it depends on actual load of concurrent agents and data traffic.

**Note:** We can’t suggest any disk space recommendations, as this highly depends on how you work. Zammad will always try to recognize the same attachments and store it just once.

3.3 Performance Tuning

As the number of active users on your system grows, performance will eventually degrade, leading to:

- delays for outgoing email,
- long loading times when viewing or creating tickets,
- stale or out-of-sync search results, or
- stale or out-of-sync ticket overviews.

You may see modest improvements by setting certain environment variables for Performance Tuning, such as $WEB_CONCURRENCY or $ZAMMAD_SESSION_JOBS_CONCURRENT.
Note:
Please ensure to meet Zammad Software requirements before hand.
This page expects administrative permissions, this is why sudo is not used.

4.1 Prerequisites

4.1.1 Additional software dependencies

In addition to already mentioned Package dependencies, some operating systems may require additional packages if not already installed.

Ubuntu / Debian

CentOS

$ apt install curl apt-transport-https gnupg

$ yum install wget epel-release

# CentOS 7
$ yum install https://download.postgresql.org/pub/repos/yum/reporpms/EL-7-x86_64/pgdg-redhat-repo-latest.noarch.rpm

4.1.2 Setup Elasticsearch

Elasticsearch is a dependency of Zammad and needs to be provided before installing Zammad. Please take a look at the following page: Set up Elasticsearch.
4.1.3 Ensure correct locale

For Zammad to function correctly, your system has to use the correct locales.

Ubuntu / Debian

CentOS

OpenSUSE / SLES

List your current locale settings.

$ locale |grep "LANG="

If above does not return <lang_code>.utf8 you can correct this issue as follows.

$ apt install locales
$ locale-gen en_US.UTF-8
$ echo "LANG=en_US.UTF-8" > /etc/default/locale

List your current locale settings.

$ locale |grep "LANG="

If above does not return <lang_code>.utf8 you can correct this issue as follows.

$ localectl set-locale LANG=en_US.utf8

List your current locale settings.

$ localectl status |grep "LC_CTYPE"

If above does not return <lang_code>.utf8 you can correct this issue as follows.

$ localectl set-locale LC_CTYPE=en_US.UTF-8

**Hint:** By default OpenSUSE uses POSIX as LANG value for the root user. Learn more about this within the OpenSUSE documentation.

This does not affect other users and thus can be ignored.

4.2 Add Repository and install Zammad

**Hint:** If you want to use MySQL instead of PostgreSQL, it’s usually enough to have the MySQL server installed on your system already. Some installation managers can’t differentiate and still install Zammad with PostgreSQL. In that case, you’ll have to adapt manually (out of scope of this documentation).

Add Repository Ubuntu

Debian

CentOS

OpenSUSE / SLES
Install Repository Key

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

Ubuntu 16.04

```bash
$ echo "deb [signed-by=/etc/apt/trusted.gpg.d/pkgr-zammad.gpg] https://dl.
    packager.io/srv/deb/zammad/zammad/stable/ubuntu 16.04 main" | \
    tee /etc/apt/sources.list.d/zammad.list > /dev/null
```

Ubuntu 18.04

```bash
$ echo "deb [signed-by=/etc/apt/trusted.gpg.d/pkgr-zammad.gpg] https://dl.
    packager.io/srv/deb/zammad/zammad/stable/ubuntu 18.04 main" | \
    tee /etc/apt/sources.list.d/zammad.list > /dev/null
```

Ubuntu 20.04

```bash
$ echo "deb [signed-by=/etc/apt/trusted.gpg.d/pkgr-zammad.gpg] https://dl.
    packager.io/srv/deb/zammad/zammad/stable/ubuntu 20.04 main" | \
    tee /etc/apt/sources.list.d/zammad.list > /dev/null
```

Install Repository Key

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

Debian 9

```bash
$ echo "deb [signed-by=/etc/apt/trusted.gpg.d/pkgr-zammad.gpg] https://dl.
    packager.io/srv/deb/zammad/zammad/stable/debian 9 main" | \
    tee /etc/apt/sources.list.d/zammad.list > /dev/null
```

Debian 10

```bash
$ echo "deb [signed-by=/etc/apt/trusted.gpg.d/pkgr-zammad.gpg] https://dl.
    packager.io/srv/deb/zammad/zammad/stable/debian 10 main" | \
    tee /etc/apt/sources.list.d/zammad.list > /dev/null
```

Debian 11

```bash
$ echo "deb [signed-by=/etc/apt/trusted.gpg.d/pkgr-zammad.gpg] https://dl.
    packager.io/srv/deb/zammad/zammad/stable/debian 11 main" | \
    tee /etc/apt/sources.list.d/zammad.list > /dev/null
```

Install Repository Key

```bash
$ rpm --import https://dl.packager.io/srv/zammad/zammad/key
```

RHEL 7 / CentOS 7

```bash
$ wget -O /etc/yum.repos.d/zammad.repo \n    https://dl.packager.io/srv/zammad/zammad/stable/installer/el/7.repo
```

RHEL 8 / CentOS 8
$ wget -O /etc/yum.repos.d/zammad.repo \

Remove obsolete Let's Encrypt CA

$ rm /usr/share/pki/trust/DST_Root_CA_X3.pem
$ update-ca-certificates

Install Repository Key

$ rpm --import https://dl.packager.io/srv/zammad/zammad/key

SLES 12 / openSUSE 42.x

$ wget -O /etc/zypp/repos.d/zammad.repo \
https://dl.packager.io/srv/zammad/zammad/stable/installer/sles/12.repo

Install Zammad Ubuntu / Debian

CentOS

OpenSUSE / SLES

$ apt update
$ apt install zammad

# CentOS 7
$ yum install postgresql14-server
$ postgresql-14-setup initdb
$ systemctl start postgresql-14
$ systemctl enable postgresql-14

# general
$ yum install zammad

Due to an issue with packager.io on CentOS 8 you’ll need to correct file permissions for public files.

chmod -R 755 /opt/zammad/public/

$ zypper ref
$ zypper install zammad

4.3 Firewall & SELinux

Some parts of these steps may not apply to you, feel free to skip them!
4.3.1 SELinux

Ubuntu / Debian / CentOS

OpenSUSE

$ # Allow nginx or apache to access public files of Zammad and communicate
$ chcon --type=httpd_sys_content_t /opt/zammad/public/
$ setsebool httpd_can_network_connect on -P
$ semanage fcontext -a -t httpd_sys_content_t /opt/zammad/public/
$ restorecon -Rv /opt/zammad/public/
$ chmod -R a+r /opt/zammad/public/

SELinux support on SUSE seems to be in early state, at least for SLES 12 (and Leap 42.x). This is why we won’t cover it in this documentation.

See the documentation for more input if you still wish to continue.

4.3.2 Firewall

Note: Below only covers the distributions default firewall. It may not cover your case.

Ubuntu

Debian

CentOS

OpenSUSE

other

$ # Open Port 80 and 443 on your Firewall
$ ufw allow 80
$ ufw allow 443
$ ufw reload

Warning: We’re covering nftables in this part - iptables is discouraged starting from Debian 10 (Buster). Our example uses the input chain, yours may be a different one!

Add the following lines to /etc/nftables.conf or your specific rule file. Ensure to add these lines to your input-chain.

# Open Port 80 and 443 for Zammad
tcp dport { http, https } accept
udp dport { http, https } accept

The result should look like the following. Keep in mind that your environment could require different / more rules.

#!/usr/local/sbin/nft -f
flush ruleset
table inet filter { (continues on next page)
chain input {
    type filter hook input priority 0; policy drop;
    ct state established,established 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;
}

To load your new rules, simply run `systemctl reload nftables`.

```
$ # Open Port 80 and 443 on your Firewall
$ firewall-cmd --zone=public --add-service=http --permanent
$ firewall-cmd --zone=public --add-service=https --permanent
$ firewall-cmd --reload
```

If your system does not yet know webserver rules, you can add a new one for your firewall by creating the file `/etc/sysconfig/SuSEfirewall2.d/services/webserver` with this content:

```
## Name: Webserver
## Description: Open ports for HTTP and HTTPS

# space separated list of allowed TCP ports
TCP="http https"
# space separated list of allowed UDP ports
UDP="http https"
```

After that locate `FW_CONFIGURATIONS_EXT` within `/etc/sysconfig/SuSEfirewall2` and add the option `webserver` to the list. The list is seperated by spaces. You may require a different zone, above covers the external zone.

Now ensure to restart the firewall service.

```
systemctl restart SuSEfirewall2
```

If we didn’t cover your distribution or firewall in question, ensure to open ports 80 and 443 (TCP & UDP) beside of the ports you need.
4.4 Manage services of Zammad

In general Zammad uses three services - these can be (re)started & stopped with the parent `zammad`.

```
$ # Zammad service to start all services at once
$ systemctl (status|start|stop|restart) zammad

$ # Zammad's internal puma server (relevant for displaying the web app)
$ systemctl (status|start|stop|restart) zammad-web

$ # Zammad's background worker - relevant for all delayed- and background jobs
$ systemctl (status|start|stop|restart) zammad-worker

$ # Zammad's websocket server for session related information
$ systemctl (status|start|stop|restart) zammad-websocket
```

4.5 Next steps

With this Zammad technically is ready to go. However, you’ll need to follow the following further steps to access Zammad’s Web-UI and getting started with it.

1. Connect Zammad with Elasticsearch
2. Configure the webserver
3. First steps
4. You may also find Zammad’s Console commands useful

If you expect usage with 5 agents or more you may also want to consider the following pages.

- Performance Tuning
- Configure Database server
CHAPTER FIVE

INSTALL FROM SOURCE

5.1 Install from source on MacOS

Note: Outdated documentation

Sorry, but this documentation part is outdated. We decided not to remove this part to provide at least hints for MacOS. Please feel welcome to provide a pull request if you find spare time!

5.1.1 Prerequisites

- Install Xcode from the App Store, open it -> Xcode menu > Preferences > Downloads -> install command line tools

```bash
$ curl -L https://get.rvm.io | bash -s stable --ruby
$ source ~/.rvm/scripts/rvm
$ start new shell -> ruby -v
```

5.1.2 Get Zammad

```bash
$ test -d ~/zammad/ || mkdir ~/zammad
$ cd ~/zammad/
```

5.1.3 Install Zammad

```bash
$ cd zammad-latest
$ bundle install
$ sudo ln -s /usr/local/mysql/lib/libmysqlclient.18.dylib /usr/lib/libmysqlclient.18.dylib # if needed!
$ rake db:create
$ rake db:migrate
$ rake db:seed
```
5.1.4 Database connect

$ cd zammad-latest
$ cp config/database/database.yml config/database.yml
$ rake db:create
$ rake db:migrate
$ rake db:seed

5.1.5 Start Zammad

$ puma -p 3000 # application web server
$ script/websocket-server.rb start # non blocking websocket server
$ script/scheduler.rb start # generate overviews on demand, just send changed data to browser

5.1.6 Visit Zammad in your browser

- http://localhost:3000/#getting_started

Note: The source installation is the most difficult installation type of Zammad. If you’re not too experienced with Linux and all that, you may want to use another installation type:

- Install from package
- Install with Docker Compose

Administrative note
Please note that we only use sudo after direct user changes. In all other situations you can expect root being in charge.

Hint: Looking for MacOS hints? You can find outdated documentation here.

5.2 Prerequisites

5.2.1 Software dependencies

Please ensure that you already provided mentioned Software requirements.
Also ensure to provide your database server and web server at this point.
5.2.2 Setup Elasticsearch

Elasticsearch is a dependency of Zammad and needs to be provided before installing Zammad. Please take a look at the following page: Set up Elasticsearch.

5.2.3 Ensure correct locale

For Zammad to function correctly, your system has to use the correct locales.

Ubuntu / Debian

CentOS

OpenSUSE / SLES

List your current locale settings.

```
$ locale |grep "LANG="
```

If above does not return `<lang_code>.utf8` you can correct this issue as follows.

```
$ apt install locales
$ locale-gen en_US.UTF-8
$ echo "LANG=en_US.UTF-8" > /etc/default/locale
```

List your current locale settings.

```
$ locale |grep "LANG="
```

If above does not return `<lang_code>.utf8` you can correct this issue as follows.

```
$ localectl set-locale LANG=en_US.utf8
```

List your current locale settings.

```
$ localectl status |grep "LC_CTYPE"
```

If above does not return `<lang_code>.utf8` you can correct this issue as follows.

```
$ localectl set-locale LC_CTYPE=en_US.UTF-8
```

**Hint:** By default OpenSUSE uses POSIX as LANG value for the root user. Learn more about this within the OpenSUSE documentation.

This does not affect other users and thus can be ignored.
5.2.4 Add user

```
$ useradd zammad -m -d /opt/zammad -s /bin/bash
$ groupadd zammad
```

5.3 Installation

5.3.1 Step 1: Get the source

**Note:** Not all distributions ship `wget` and `tar` by default, you may need to install it manually.


```
$ cd /opt
$ wget https://github.com/zammad/zammad/archive/stable.tar.gz
$ tar -xzf stable.tar.gz --strip-components 1 -C zammad
$ chown -R zammad:zammad zammad/
$ rm -f stable.tar.gz
```

5.3.2 Step 2: Install dependencies

**Note:**

**Below commands do neither include the database server nor the web server.**

We do cover important web server related stuff within *Configure the webserver.*

Zammad requires specific ruby versions. Adapt the commands below if you install older versions. A list of required versions can be found on the *Software requirements* page.

Ubuntu

Debian

CentOS

OpenSuSE

other

**Install Node.js**

```
$ apt update
$ apt install curl
$ curl -fsSL https://deb.nodesource.com/setup_lts.x | bash -
$ apt install nodejs
```

**Install RVM**
Zammad

$ apt install curl git patch build-essential bison zlib1g-dev libssl-dev libxml2-dev libxml2-1-dev libxslt1-dev libyaml-dev autoconf automake libreadline-dev libyaml-dev libtool pkg-config libffi-dev libimlib2-dev gawk libsqlite3-dev sqlite3 software-properties-common

$ apt-add-repository -y ppa:rael-gc/rvm
$ apt update
$ apt install rvm

Set relevant Environment variables

```bash
# Set rails environment specific things
$ echo "export RAILS_ENV=production" >> /opt/zammad/.bashrc
$ echo "export RAILS_SERVE_STATIC_FILES=true" >> /opt/zammad/.bashrc
$ echo "rvm --default use 3.0.4" >> /opt/zammad/.bashrc

# Debian, CentOS & OpenSuSE
$ echo "source /usr/local/rvm/scripts/rvm" >> /opt/zammad/.bashrc
# Ubuntu
$ echo "source /usr/share/rvm/scripts/rvm" >> /opt/zammad/.bashrc
```

Install Ruby Environment

```bash
# Add zammad user to RVM group
$ usermod -a -G rvm zammad

# Install Ruby 3.0.4
$ su - zammad
$ rvm install ruby-3.0.4

# Install bundler, rake and rails
$ rvm use 3.0.4
$ gem install bundler rake rails
```

Install Node.js

```bash
$ apt update
$ apt install curl
$ curl -fsSL https://deb.nodesource.com/setup_lts.x | bash -
$ apt install nodejs
```

Install RVM

```bash
$ apt install curl git patch build-essential bison zlib1g-dev libssl-dev libxml2-dev libxml2-1-dev libxslt1-dev libyaml-dev autoconf automake libreadline-dev libyaml-dev libtool pkg-config libffi-dev libimlib2-dev gawk libsqlite3-dev sqlite3

$ gpg --keyserver keyserver.ubuntu.com --recv-keys
409B6B1796C275462A1703113804BB82D39DC0E3 7D28BF1CF37B13E2069D6956105BD0E739499DB
$ curl -L https://get.rvm.io | bash -s stable
```

5.3. Installation
Zammad

Set relevant Environment variables

```bash
# Set rails environment specific things
$ echo "export RAILS_ENV=production" >> /opt/zammad/.bashrc
$ echo "export RAILS_SERVE_STATIC_FILES=true" >> /opt/zammad/.bashrc
$ echo "rvm --default use 3.0.4" >> /opt/zammad/.bashrc

# Debian, CentOS & OpenSuSE
$ echo "source /usr/local/rvm/scripts/rvm" >> /opt/zammad/.bashrc

# Ubuntu
$ echo "source /usr/share/rvm/scripts/rvm" >> /opt/zammad/.bashrc
```

Install Ruby Environment

```bash
# Add zammad user to RVM group
$ usermod -a -G rvm zammad

# Install Ruby 3.0.4
$ su - zammad
$ rvm install ruby-3.0.4

# Install bundler, rake and rails
$ rvm use 3.0.4
$ gem install bundler rake rails
```

Install Node.js

```bash
$ curl -fsSL https://rpm.nodesource.com/setup_lts.x | bash -
```

Install RVM

```bash
$ yum install epel-release
$ yum install patch autoconf automake bison bzip2 gcc-c++ libffi-devel libtool make.
--patch readline-devel ruby sqlite-devel\n  zlib-devel glibc-headers glibc-devel openssl-devel git imlib2 imlib2-devel
$ gpg --keyserver keyserver.ubuntu.com --recv-keys\n  409B6B1796C275462A1703113B04BB82D39DC0E3 7D2B8AF1CF37B13E2069D6956105BD0E739499BDB
$ curl -L https://get.rvm.io | bash -s stable
```

Set relevant Environment variables

```bash
# Set rails environment specific things
$ echo "export RAILS_ENV=production" >> /opt/zammad/.bashrc
$ echo "export RAILS_SERVE_STATIC_FILES=true" >> /opt/zammad/.bashrc
$ echo "rvm --default use 3.0.4" >> /opt/zammad/.bashrc

# Debian, CentOS & OpenSuSE
$ echo "source /usr/local/rvm/scripts/rvm" >> /opt/zammad/.bashrc

# Ubuntu
$ echo "source /usr/share/rvm/scripts/rvm" >> /opt/zammad/.bashrc
```

Install Ruby Environment

```bash
```
# Add zammad user to RVM group
$ usermod -a -G rvm zammad

# Install Ruby 3.0.4
$ su - zammad
$ rvm install ruby-3.0.4

# Install bundler, rake and rails
$ rvm use 3.0.4
$ gem install bundler rake rails

Install Node.js

$ zypper install nodejs16

Install RVM

$ zypper install patch autoconf automake bison bzip2 gcc-c++ libffi-devel libtool
  --make patch readline-devel sqlite3-devel
  sqlite3 zlib-devel glibc-devel openssl-devel git imlib2 imlib2-devel gdbm-devel
  --libyaml-devel

$ gpg --keyserver keyserver.ubuntu.com --recv-keys
  4096B6B1796C275462A1703113804BB82D39DC0E3 7D2BAF1CF37B13E2069D6956105BD0E739499BDB
$ curl -L https://get.rvm.io | bash -s stable

Set relevant Environment variables

# Set rails environment specific things
$ echo "export RAILS_ENV=production" >> /opt/zammad/.bashrc
$ echo "export RAILS_SERVE_STATIC_FILES=true" >> /opt/zammad/.bashrc
$ echo "rvm --default use 3.0.4" >> /opt/zammad/.bashrc

# Debian, CentOS & OpenSuSE
$ echo "source /usr/local/rvm/scripts/rvm" >> /opt/zammad/.bashrc
# Ubuntu
$ echo "source /usr/share/rvm/scripts/rvm" >> /opt/zammad/.bashrc

Install Ruby Environment

# Add zammad user to RVM group
$ usermod -a -G rvm zammad

# Install Ruby 3.0.4
$ su - zammad
$ rvm install ruby-3.0.4

# Install bundler, rake and rails
$ rvm use 3.0.4
$ gem install bundler rake rails

Other systems than above mentioned are out of scope of this documentation. Please check the rvm documentation on how to install rvm on your system.

Please also ensure to install nodejs.
After that install the specific required ruby version.

After installing bundler, rake and rails we’ll need to install all required gems. The command depends on the database server you are using.

**PostgreSQL (recommended)**

**MySQL / MariaDB**

### Install PostgreSQL Dependencies

**Ubuntu / Debian**

- CentOS
- OpenSuSE

```
$ apt install libpq-dev
```

# CentOS 7
```
$ yum install -y https://download.postgresql.org/pub/repos/yum/reporpms/EL-7-x86_64/\n-->pgdg-redhat-repo-latest.noarch.rpm
$ yum install postgresql13-libs postgresql13-devel
```

# CentOS 8
```
$ yum install postgresql-libs postgresql-devel
```

```
$ zypper install postgresql-devel
```

### Install Gems for Zammad

```
$ su - zammad
$ bundle config set without "test development mysql"
$ bundle install
```

# CentOS 7 users - above command might fail, run the following

# command and repeat above bundle install.

# Adjust pg_config path according to your environment
```
$ gem install pg -v '0.21.0' -- --with-pg-config=/usr/pgsql-13/bin/pg_config
```

### Install MySQL/MariaDB Dependencies

**Ubuntu / Debian**

- CentOS
- OpenSuSE

```
$ apt install libmariadb-dev
```

```
$ yum install mariadb-devel
```

```
$ zypper install libmariadb-devel
```

### Create database

While that’s basically an easy step, you may want to create your database as follows to minimize potential issues.

Below commands need adjustments to fit your environment. Choose a safe password.
$ mysql
> create user 'zammad'@'localhost' identified by 'changeme';
> CREATE DATABASE zammad DEFAULT CHARACTER SET utf8 DEFAULT COLLATE utf8_general_ci;
> GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, INDEX, DROP, ALTER, CREATE...
   → TEMPORARY TABLES,
   → LOCK TABLES ON zammad.* TO 'zammad'@'localhost';

Install Gems for Zammad

$ su - zammad
$ bundle set config without "test development postgres"
$ bundle install

5.3.3 Step 3: Configure database settings

Tip: For easiest usage ...
If you provide your Zammad user with database creation permission, you can run the step 4 without adjustment. If you don’t want that, you’ll have to create the database manually.

$ cp config/database/database.yml config/database.yml
$ vi config/database.yml

Here’s a sample configuration to give you an idea on how your configuration file could look like. Please also have a look at Configure Database server for deeper details.

PostgreSQL
MySQL / MariaDB

production:
  adapter: postgresql
  database: zammad
  pool: 50
  timeout: 5000
  encoding: utf8
  username: zammad
  password: changeme

Hint: You can remove the password line if you enable socket based authentication!

production:
  adapter: mysql2
  database: zammad
  pool: 50
  timeout: 5000
  encoding: utf8
  username: zammad
  password: changeme

5.3. Installation
**Hint:** If you want to use an existing database server that’s not on the same machine, you can also use host and port to set that up.

For security reasons, ensure that your database configuration is readable for the Zammad user only.

```
$ chmod 600 /opt/zammad/config/database.yml
$ chown zammad:zammad /opt/zammad/config/database.yml
```

### 5.3.4 Step 4: Initialize your database

**Warning:** Ensure to do this as zammad user in your Zammad directory!

```
$ su - zammad
$ rake db:create  # SKIP IF you already created zammads database (see tip of step 3)
$ rake db:migrate
$ rake db:seed
# Synchronize translations
$ rails r "Locale.sync"
$ rails r "Translation.sync"
```

### 5.3.5 Step 5: Pre compile all Zammad assets

```
$ rake assets:precompile
```

### 5.3.6 Step 6: Start Zammad or install as service

**Note:** Run the following commands as root.

You can start all services by hand or use systemd to start / stop Zammad. systemd (recommended)

the manual way

```
$ cd /opt/zammad/script/systemd
$ ./install-zammad-systemd-services.sh
```

**Warning:** This method is not suitable for production use - you should avoid it.

```
$ rails s -p 3000  # application web server
$ script/websocket-server.rb start  # non blocking websocket server
$ script/background-worker.rb start  # generate overviews on demand, just send changed data to browser
```
5.4 Manage services of Zammad

In general Zammad uses three services - these can be (re)started & stopped with the parent zammad.

```
$ # Zammad service to start all services at once
$ systemctl (status|start|stop|restart) zammad

$ # Zammad's internal puma server (relevant for displaying the web app)
$ systemctl (status|start|stop|restart) zammad-web

$ # Zammad's background worker - relevant for all delayed- and background jobs
$ systemctl (status|start|stop|restart) zammad-worker

$ # Zammad's websocket server for session related information
$ systemctl (status|start|stop|restart) zammad-websocket
```

5.5 Firewall & SELinux

Some parts of these steps may not apply to you, feel free to skip them!

5.5.1 SELinux

Ubuntu / Debian / CentOS

OpenSUSE

```
$ # Allow nginx or apache to access public files of Zammad and communicate
$ chcon -Rv --type=httpd_sys_content_t /opt/zammad/public/
$ setsebool httpd_can_network_connect on -P
$ semanage fcontext -a -t httpd_sys_content_t /opt/zammad/public/
$ restorecon -Rv /opt/zammad/public/
$ chmod -R a+r /opt/zammad/public/
```

SELinux support on SUSE seems to be in early state, at least for SLES 12 (and Leap 42.x). This is why we won't cover it in this documentation.

See the documentation for more input if you still wish to continue.
5.5.2 Firewall

**Note:** Below only covers the distributions default firewall. It may not cover your case.

Ubuntu
Debian
CentOS
OpenSUSE
other

```
$ # Open Port 80 and 443 on your Firewall
$ ufw allow 80
$ ufw allow 443
$ ufw reload
```

**Warning:** We’re covering `nftables` in this part - `iptables` is discouraged starting from Debian 10 (Buster). Our example uses the `input` chain, yours may be a different one!

Add the following lines to `/etc/nftables.conf` or your specific rule file. Ensure to add these lines to your input-chain.

```
# Open Port 80 and 443 for Zammad
tcp dport { http, https } accept
udp dport { http, https } accept
```

The result should look like the following. Keep in mind that your environment could require different / more rules.

```
#!/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;
    }
}
```

To load your new rules, simply run `systemctl reload nftables`. 
$ # Open Port 80 and 443 on your Firewall
$ firewall-cmd --zone=public --add-service=http --permanent
$ firewall-cmd --zone=public --add-service=https --permanent
$ firewall-cmd --reload

If your system does not yet know webserver rules, you can add a new one for your firewall by creating the file `/etc/sysconfig/SuSEfirewall2.d/services/webserver` with this content:

```bash
## Name: Webserver
## Description: Open ports for HTTP and HTTPS

# space separated list of allowed TCP ports
TCP="http https"

# space separated list of allowed UDP ports
UDP="http https"
```

After that locate `FW_CONFIGURATIONS_EXT` within `/etc/sysconfig/SuSEfirewall2` and add the option `webserver` to the list. The list is seperated by spaces. You may require a different zone, above covers the external zone.

Now ensure to restart the firewall service.

```bash
systemctl restart SuSEfirewall2
```

If we didn’t cover your distribution or firewall in question, ensure to open ports 80 and 443 (TCP & UDP) beside of the ports you need.

### 5.6 Next steps

With this Zammad technically is ready to go. However, you’ll need to follow the following further steps to access Zammads Web-UI and getting started with it.

1. **Connect Zammad with Elasticsearch**
2. **Configure the webserver**
3. **First steps**
4. You may also find Zammads `Console` commands useful

If you expect usage with 5 agents or more you may also want to consider the following pages.

- **Performance Tuning**
- **Configure Database server**
Zammad’s search function is powered by Elasticsearch, and requires the ingest attachment plugin.

**Note:** This guide uses the `zammad run` command prefix in command line examples. This prefix is only applicable to package installations (i.e., via apt/yum/zypper, or .deb/.rpm files).

If you installed from source, be sure to omit this prefix and run the bare `rails` ... or `rake` ... commands instead.

### 6.1 Step 1: Installation

**Note:** To use OSS or not to use...

Starting with Zammad 4.0 our packages allow you to decide whether to use `elasticsearch` or `elasticsearch-oss`.

`elasticsearch-oss` users please use below “direct download” tab for further installation steps.

**Warning:** Above does not apply to CentOS because of compatibility reasons.

- Ubuntu
- Debian
- CentOS
- OpenSUSE

**Direct Download**

```bash
$ 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
```
Zammad

$ 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

$ rpm --import https://artifacts.elastic.co/GPG-KEY-elasticsearch
$ echo "[[elasticsearch-7.x]
name=Elasticsearch repository for 7.x packages
baseurl=https://artifacts.elastic.co/packages/7.x/yum
gpgcheck=1
gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch
enabled=1
autorefresh=1
type=rpm-md"| tee /etc/yum.repos.d/elasticsearch-7.x.repo
$ yum install -y elasticsearch
$ /usr/share/elasticsearch/bin/elasticsearch-plugin install ingest-attachment

# Install the attachment plugin
$ /usr/share/elasticsearch/bin/elasticsearch-plugin install ingest-attachment

# Increase the virtual memory map limit
$ sysctl -w vm.max_map_count=262144

After you installed Elasticsearch and its attachment plugin, ensure to enable it by default and start it.

$ systemctl start elasticsearch
$ systemctl enable elasticsearch

Note: Docker installations on macOS/Windows:
6.2 Step 2: Suggested Configuration

We use the following settings to optimize the performance of our Elasticsearch servers. Your mileage may vary.

```yaml
# /etc/elasticsearch/elasticsearch.yml

# Tickets above this size (articles + attachments + metadata)
# may fail to be properly indexed (Default: 100mb).
#
# When Zammad sends tickets to Elasticsearch for indexing,
# it bundles together all the data on each individual ticket
# and issues a single HTTP request for it.
# Payloads exceeding this threshold will be truncated.
#
# Performance may suffer if it is set too high.
http.max_content_length: 400mb

# Allows the engine to generate larger (more complex) search queries.
# Elasticsearch will raise an error or deprecation notice if this value is too low,
# but setting it too high can overload system resources (Default: 1024).
#
# Available in version 6.6+ only.
indices.query.bool.max_clause_count: 2000
```

Note: For more information on the `indices.query.bool.max_clause_count` setting, see the Elasticsearch 6.6 release notes.

6.3 Step 3: Connect Zammad

Hint: Before proceeding here...

Make sure to install Zammad before running below commands, as this will fail otherwise.

- install from `package`
- install from `source`

```bash
# Set the Elasticsearch server address
$ zammad run rails r "Setting.set('es_url', 'http://localhost:9200')"

# Build the search index
$ zammad run rake zammad:searchindex:rebuild
```
6.3.1 Optional settings

Authentication
Index namespaces
File-attachment indexing rules

```
# HTTP Basic
$ zammad run rails r "Setting.set('es_user', '<username>')"
$ zammad run rails r "Setting.set('es_password', '<password>')"
```

Hint: How do I set up authentication on my Elasticsearch server?

Elasticsearch provides many different authentication methods. Some of them may require paid X-Pack, please check the elastic documentation for more information.

Useful when connecting multiple services or Zammad instances to a single Elasticsearch server (to prevent name collisions during indexing).

```
$ zammad run rails r "Setting.set('es_index', Socket.gethostname.downcase + '_zammad')"
```

Zammad supports searching by the contents of file attachments, which means Elasticsearch has to index those, too. Limiting such indexing can help conserve system resources.

```
# Files with these extensions will not be indexed

# Files larger than this size (in MB) will not be indexed
$ zammad run rails r "Setting.set('es_attachment_max_size_in_mb', 50)"
```

6.4 Appendix

6.4.1 List of Indexed Attributes

Below is a comprehensive list of all object attributes indexed by Elasticsearch. In other words, if you wish to find a ticket, article, or user via the Zammad search box, Elasticsearch can match on any (or all) of the fields below.

Table of content

- Ticket
- Ticket Priority
- Ticket State
- Article
- User
- Organization
Note: These fields may vary if you created custom fields (objects) in the admin interface.

Warning: Zammad 4.0 introduced breaking changes on the Elasticsearch index.

Hint: Below list contains functionality hints
In order to save space and duplicate information, we’ll provide hints to functions within brackets if applicable.

• (SLA):
  Attributes marked as SLA attribute are only set if the ticket is affected by SLA calculation. Please note that some attributes may not be set if specific conditions are not met.
  Also note that some attributes may be reset to null if no longer applicable.

• note attribute: Note attributes usually are empty if not specified via console or API.

• Timestamps: All timestamps provided by Zammad are UTC by default. This also applies to times provided by Elasticsearch

Ticket

Tip: The following indice contains below mentioned information: *_ticket

<table>
<thead>
<tr>
<th>Field</th>
<th>Sample Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>article</td>
<td>#{Article Array}</td>
<td>Array with all articles belonging to the ticket</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Please see Article for more details</td>
</tr>
<tr>
<td>article_count</td>
<td>1</td>
<td>Number of articles within the ticket</td>
</tr>
<tr>
<td>close_at</td>
<td>null, 2021-03-03T14:50:20.673Z</td>
<td>First close time, set once</td>
</tr>
<tr>
<td>close_diff_in_min</td>
<td>null, 239, -5</td>
<td>Depends on close_in_min and tells how many minutes the ticket was closed relative to SLAs solution time. (SLA)</td>
</tr>
<tr>
<td>close_escalation_at</td>
<td>null, 2021-03-03T15:50:20.673Z</td>
<td>Time stamp when the ticket would escalate in case solution time is violated. (SLA)</td>
</tr>
<tr>
<td>close_in_min</td>
<td>null, 11</td>
<td>Value in minutes for how long the ticket was open based on business hours. (SLA)</td>
</tr>
</tbody>
</table>

continues on next page
<table>
<thead>
<tr>
<th>Field</th>
<th>Sample Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>create_article_sender</td>
<td>Contains these attributes:</td>
<td>Sender of the article (System, Agent, Customer)</td>
</tr>
<tr>
<td></td>
<td>• note: null</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• updated_at: 2021-03-03T14:50:20.812Z</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• name: Customer</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• created_at: 2021-03-03T14:50:20.812Z</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• updated_by_id: 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• id: 2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• created_by_id: 1</td>
<td></td>
</tr>
<tr>
<td>create_article_sender_id</td>
<td>1, 2</td>
<td>ID of the user that created the article</td>
</tr>
<tr>
<td>create_article_type</td>
<td>Contains these attributes:</td>
<td>Information of first article type and nature</td>
</tr>
<tr>
<td></td>
<td>• note: null</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• updated_at: 2021-03-03T14:50:20.812Z</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• name: phone, email, web</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• active: true</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• created_at: 2021-03-03T14:50:20.812Z</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• updated_by_id: 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• id: 5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• created_by_id: 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• communication: true, false</td>
<td></td>
</tr>
<tr>
<td>create_article_type_id</td>
<td>5</td>
<td>Type ID of first article</td>
</tr>
<tr>
<td>created_at</td>
<td>2021-03-24T16:17:27.210Z</td>
<td>Time stamp of ticket creation</td>
</tr>
<tr>
<td>created_by</td>
<td>#{user object}</td>
<td>Complete Payload of user that created the ticket</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Please see User for more</td>
</tr>
<tr>
<td>created_by_id</td>
<td>3</td>
<td>User ID that created the ticket</td>
</tr>
<tr>
<td>customer</td>
<td>#{user object}</td>
<td>Complete payload of the customer that created the ticket</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Please see User for more</td>
</tr>
<tr>
<td>customer_id</td>
<td>8</td>
<td>Customers User ID</td>
</tr>
</tbody>
</table>

continues on next page
Table 1 – continued from previous page

<table>
<thead>
<tr>
<th>Field</th>
<th>Sample Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>escalation_at</td>
<td>null, 2021-03-24T16:28:38.535Z</td>
<td>Time stamp of the next applicable escalation. One of the following attributes:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• close_escalation_at</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• first_response_escalation_at</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• update_escalation_at (SLA)</td>
</tr>
<tr>
<td>first_response_at</td>
<td>null, 2021-03-24T16:28:38.303Z</td>
<td>Time stamp of the first communication type reaction to the customer (SLA)</td>
</tr>
<tr>
<td>first_response_diff_in_min</td>
<td>null, 10, -6</td>
<td>Depends on first_response_in_min and tells how many minutes the tickets first response took relative to the first response time of your SLA. (SLA)</td>
</tr>
<tr>
<td>first_response_in_min</td>
<td>null, 11</td>
<td>Value in minutes for how long the first response took based on the business hours. (SLA)</td>
</tr>
<tr>
<td>group</td>
<td>#{group object}</td>
<td>Complete payload of the current tickets group</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Please see Group for more</td>
</tr>
<tr>
<td>group_id</td>
<td>1</td>
<td>ID of the current group</td>
</tr>
<tr>
<td>id</td>
<td>1, 111</td>
<td>ID of the Ticket</td>
</tr>
<tr>
<td>last_contact_agent_at</td>
<td>null, 2021-03-24T16:28:38.303Z</td>
<td>Time stamp of last communication type contact of any agent</td>
</tr>
<tr>
<td>last_contact_at</td>
<td>null, 2021-03-24T16:28:38.303Z</td>
<td>Time stamp of last communication type contact</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Depends on last_contact_agent_at, last_contact_customer_at and “Ticket Last Contact Behaviour” setting</td>
</tr>
<tr>
<td>last_contact_customer_at</td>
<td>null, 2021-03-24T16:28:38.303Z</td>
<td>Time stamp of last communication type contact of customer</td>
</tr>
<tr>
<td>mention_user_ids</td>
<td>[3, 5]</td>
<td>Array with mentioned or subscribed users IDs</td>
</tr>
<tr>
<td>note</td>
<td>null</td>
<td>Note of ticket, only set via console or API</td>
</tr>
<tr>
<td>number</td>
<td>1001038, 202006231010138</td>
<td>Ticket number</td>
</tr>
<tr>
<td>organization</td>
<td>null, #{organization object}</td>
<td>Complete Payload of user that owns the ticket</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Please see Organization for more</td>
</tr>
<tr>
<td>organization_id</td>
<td>null, 2</td>
<td>ID of the customers organization</td>
</tr>
<tr>
<td>owner</td>
<td>null, #{user object}</td>
<td>Complete Payload of user that owns the ticket</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Please see User for more</td>
</tr>
<tr>
<td>owner_id</td>
<td>null, 3</td>
<td>User ID of the ticket owner</td>
</tr>
</tbody>
</table>

continues on next page
<table>
<thead>
<tr>
<th>Field</th>
<th>Sample Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>pending_time</td>
<td>null, 2021-03-24T17:44:06.912Z</td>
<td>Depends on pending states, time stamp for pending time</td>
</tr>
<tr>
<td>preferences</td>
<td>n/a, special information for internal functions</td>
<td>May not be available in your system, contains information for internal system functions</td>
</tr>
<tr>
<td>priority</td>
<td>#{priority object}</td>
<td>Complete Payload of priority of ticket Please see Ticket Priority for more</td>
</tr>
<tr>
<td>priority_id</td>
<td>2</td>
<td>Priority ID of the ticket</td>
</tr>
<tr>
<td>state</td>
<td>#{state object}</td>
<td>Complete Payload of current ticket state Please see Ticket State for more</td>
</tr>
<tr>
<td>state_id</td>
<td>1, 4</td>
<td>ID of current ticket state</td>
</tr>
<tr>
<td>tags</td>
<td>[&quot;order&quot;, &quot;americano&quot;]</td>
<td>Array with all attached tags</td>
</tr>
<tr>
<td>time_unit</td>
<td>null, 15</td>
<td>Accounted time units for ticket (total)</td>
</tr>
<tr>
<td>title</td>
<td>Feedback Form, Need help</td>
<td>Title / Subject of Ticket</td>
</tr>
<tr>
<td>type</td>
<td>null</td>
<td>Ticket type (deprecated)</td>
</tr>
<tr>
<td>update_diff_in_min</td>
<td>null, 2021-03-24T16:28:38.303Z</td>
<td>Depends on update_in_min and tells how many minutes the last ticket update took relatively to the update time setting (SLA)</td>
</tr>
<tr>
<td>update_escalation_at</td>
<td>null, 2021-03-24T16:28:38.303Z</td>
<td>Time stamp when the ticket would escalate in case update time is violated. (SLA)</td>
</tr>
<tr>
<td>update_in_min</td>
<td>null, 5, -10</td>
<td>Value in minutes for how long the last ticket update took based on the business hours and update time. (SLA)</td>
</tr>
<tr>
<td>updated_at</td>
<td>2021-03-24T16:28:38.303Z</td>
<td>Last ticket update</td>
</tr>
<tr>
<td>updated_by</td>
<td>#{user object}</td>
<td>Complete Payload of the user that updated the ticket Please see User for more</td>
</tr>
<tr>
<td>updated_by_id</td>
<td>1, 3</td>
<td>User ID that updated the ticket</td>
</tr>
</tbody>
</table>

**Ticket Priority**

Tip: The following indice contains below mentioned information: *._ticket_priority*
### Table 2: Ticket Priority-Index

<table>
<thead>
<tr>
<th>Field</th>
<th>Sample Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>active</td>
<td>true, false</td>
<td>Defines if the priority is active (available)</td>
</tr>
<tr>
<td>created_at</td>
<td>2021-03-03T14:50:20.724Z</td>
<td>Creation date of priority</td>
</tr>
<tr>
<td>created_by_id</td>
<td>1</td>
<td>User that created priority</td>
</tr>
<tr>
<td>default_create</td>
<td>false, true</td>
<td>Defines if priority is default priority upon ticket creation</td>
</tr>
<tr>
<td>id</td>
<td>3</td>
<td>ID of priority</td>
</tr>
<tr>
<td>name</td>
<td>3 high</td>
<td>Priority name</td>
</tr>
<tr>
<td>note</td>
<td>null</td>
<td>Note for priority that has been set via console or API</td>
</tr>
<tr>
<td>ui_color</td>
<td>null, high-priority</td>
<td>CSS class for tickets of priority</td>
</tr>
<tr>
<td>ui_icon</td>
<td>null, important</td>
<td>CSS class for ticket icons of priority</td>
</tr>
<tr>
<td>updated_at</td>
<td>2021-03-03T14:50:20.724Z</td>
<td>Date of last change</td>
</tr>
<tr>
<td>updated_by_id</td>
<td>1</td>
<td>User ID of user last updating the priority</td>
</tr>
</tbody>
</table>

**Ticket State**

**Tip:** The following index contains below mentioned information: `_ticket_state_`
Table 3: Ticket State-Index

<table>
<thead>
<tr>
<th>Field</th>
<th>Sample Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>active</td>
<td>true, false</td>
<td>Defines if state is active (available)</td>
</tr>
<tr>
<td>created_at</td>
<td>2021-03-03T14:50:20.694Z</td>
<td>Creation date</td>
</tr>
<tr>
<td>created_by_id</td>
<td>1</td>
<td>User ID that created state</td>
</tr>
<tr>
<td>default_create</td>
<td>false, true</td>
<td>Defines if the state is the default state upon ticket creation</td>
</tr>
<tr>
<td>default_follow_up</td>
<td>false, true</td>
<td>Defines if the state is the default follow up state on ticket follow ups</td>
</tr>
<tr>
<td>id</td>
<td>7</td>
<td>State ID</td>
</tr>
<tr>
<td>ignore_escalation</td>
<td>false, true</td>
<td>Defines if SLA calculation is generally ignored for this state</td>
</tr>
<tr>
<td>name</td>
<td>pending close</td>
<td>State name</td>
</tr>
<tr>
<td>next_state</td>
<td>n/a, #{state object}</td>
<td>Contains all follow up state information if applicable, may not be available depending on the state type</td>
</tr>
<tr>
<td>next_state_id</td>
<td>null, 4</td>
<td>State ID of follow up state</td>
</tr>
<tr>
<td>note</td>
<td>null</td>
<td>Note that has been set via console or API</td>
</tr>
</tbody>
</table>

**state_type**

Contains these attributes:
- created_at: 2021-03-03T14:50:20.582Z
- created_by_id: 1
- id: 4
- name: pending action
- note: null
- updated_at: 2021-03-03T14:50:20.582Z
- updated_by_id: 1

Contains all available information of the states type

<table>
<thead>
<tr>
<th>field</th>
<th>Sample Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>state_type_id</td>
<td>4</td>
<td>ID of the state type</td>
</tr>
<tr>
<td>updated_at</td>
<td>2021-03-03T14:50:20.694Z</td>
<td>Last update of state</td>
</tr>
<tr>
<td>updated_by_id</td>
<td>1</td>
<td>User ID that updated state last</td>
</tr>
</tbody>
</table>

**Article**

**Tip:** The following indice contains below mentioned information: *_ticket

**Note:** Articles are part of the ticket index. To reduce complexity we decided to provide it in its own table.
### Table 4: Article-Index

<table>
<thead>
<tr>
<th>Field</th>
<th>Sample Value</th>
<th>Description</th>
</tr>
</thead>
</table>
| body           | Hi, \n
please send me: \n
75007 Paris \n
David Bell | Article body in plain text                            |
| cc             | null, alias@domain.tld                                  | EMail-Addresses set as CC (String)               |
| content_type   | text/html                                               | Content type of article                           |
| created_at     | 2021-03-22T03:47:59.290Z                                | Time stamp of article creation                   |
| created_by_id  | 10                                                      | User ID that created the article                  |
| from           | David Bell <david@example.com>                          | From field of article creator                    |
| id             | 16                                                      | Internal article ID                              |
| in_reply_to    | null                                                    | In-Reply-To Header from emails if applicable     |
| internal       | false, true                                             | Defines if article is internal                    |
| message_id     | null                                                    | Message ID of Email if applicable                |
| origin_by_id   | null                                                    | User ID or original creator if created on behalf another user |
| preferences    | {}                                                      | Internal preferences, may be empty, mainly for delivery states |
| references     | null                                                    | Contains message references                      |
| reply_to       | null                                                    | Contains reply to header if applicable           |
| sender_id      | 2                                                       | ID of sender type (Customer, System, Agent)      |
| subject        | My amazing subject                                      | Article subject                                  |
| ticket_id      | 9                                                       | Ticket ID the article belongs to                  |
| to             | support@example.com                                     | EMail address from TO-Header                      |
| type_id        | 1                                                       | ID of articles Type (phone, email, web, ...)      |
| updated_at     | 2021-03-22T03:47:59.290Z                                | Last update                                      |
| updated_by_id  | 10                                                      | User that updated article                        |

### User

**Tip:** The following indice contains below mentioned information: *_user*

### Table 5: User-Index

<table>
<thead>
<tr>
<th>Field</th>
<th>Sample Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>active</td>
<td>true, false</td>
<td>Defines if user is active</td>
</tr>
</tbody>
</table>
| address       | Benelong Point

Sydney NSW 2000 | Address string                                       |
| city          | Berlin                                                 | City string                                      |
| country       | Germany                                                | Country string                                   |
| created_at    | 2021-03-22T12:47:56.460Z                               | Creation date of user                            |
| created_by_id | 1                                                      | User ID that created the user                     |
| department    | IT                                                     | Department string                                |
| email         | alias@domain.tld                                       | EMail Address of user, if applicable             |
| fax           | 1234                                                   | Fax number                                       |
| firstname     | null, John                                             | Users first name                                 |

continues on next page
Table 5 – continued from previous page

<table>
<thead>
<tr>
<th>Field</th>
<th>Sample Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>8</td>
<td>Internal User ID</td>
</tr>
<tr>
<td>last_login</td>
<td>null, 2021-03-23T12:47:56.460Z</td>
<td>Updated upon every user login</td>
</tr>
<tr>
<td>lastname</td>
<td>null, Doe</td>
<td>Users last name</td>
</tr>
<tr>
<td>login</td>
<td>auto-1234567, jdoe</td>
<td>Login name, always set and unique, can differ from email</td>
</tr>
<tr>
<td>mobile</td>
<td>&quot;&quot;, 1232</td>
<td>Mobile phone number</td>
</tr>
<tr>
<td>note</td>
<td>&quot;&quot;</td>
<td>Note being available via web, console and API</td>
</tr>
<tr>
<td>organization</td>
<td>#{organization object}</td>
<td>Complete Payload of the organization the user is member of Please see Organization for more</td>
</tr>
<tr>
<td>organization_id</td>
<td>3</td>
<td>ID of organization the user is member of</td>
</tr>
<tr>
<td>out_of_office</td>
<td>false, true</td>
<td>Defines if user has activated out of office function</td>
</tr>
<tr>
<td>out_of_office_end_at</td>
<td>null, 2021-03-26</td>
<td>Ending date out of office</td>
</tr>
<tr>
<td>out_of_office_replacement</td>
<td>null, 3</td>
<td>User ID that replaces this user during out of office period</td>
</tr>
<tr>
<td>out_of_office_start_at</td>
<td>null, 2021-03-24</td>
<td>Begin date out of office</td>
</tr>
<tr>
<td>permissions</td>
<td>(Array)</td>
<td>Array with all permissions of the user</td>
</tr>
<tr>
<td>phone</td>
<td>&quot;&quot;, 0061 2 1234 7777</td>
<td>Phone number of user</td>
</tr>
<tr>
<td>preferences</td>
<td>{}, #{several preference attributes}</td>
<td>Depends on user and situation, may contain notification_config, locale and other internal system information</td>
</tr>
<tr>
<td>role_ids</td>
<td>(Array), [1, 2]</td>
<td>Contains array with role IDs assigned to the user</td>
</tr>
<tr>
<td>street</td>
<td>&quot;&quot;</td>
<td>Street</td>
</tr>
<tr>
<td>updated_at</td>
<td>2021-03-25T00:27:52.308Z</td>
<td>Time stamp of last update</td>
</tr>
<tr>
<td>updated_by_id</td>
<td>3</td>
<td>User ID that updated this entry</td>
</tr>
<tr>
<td>verified</td>
<td>false, true</td>
<td>Defines if the user has verified the account</td>
</tr>
<tr>
<td>vip</td>
<td>false, true</td>
<td>Defines if user has VIP state</td>
</tr>
<tr>
<td>web</td>
<td>&quot;&quot;, <a href="https://zammad.org">https://zammad.org</a></td>
<td>Web URL of User</td>
</tr>
<tr>
<td>zip</td>
<td>&quot;&quot;, 10123</td>
<td>ZIP code</td>
</tr>
</tbody>
</table>

**Organization**

**Tip:** The following indice contains below mentioned information: *_organization*
### Table 6: Organization-Index

<table>
<thead>
<tr>
<th>Field</th>
<th>Sample Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>active</td>
<td><code>true, false</code></td>
<td>Defines if organization is active</td>
</tr>
<tr>
<td>created_at</td>
<td>2021-03-22T12:47:54.807Z</td>
<td>Creation date</td>
</tr>
<tr>
<td>created_by</td>
<td>#{user object}</td>
<td>Complete Payload of the user that created the organization</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Please see <a href="#">User</a> for more</td>
</tr>
<tr>
<td>created_by_id</td>
<td>1</td>
<td>User ID that created the organization</td>
</tr>
<tr>
<td>domain</td>
<td><code>null, example.com</code></td>
<td>Organizations domain</td>
</tr>
<tr>
<td>domain_assignment</td>
<td><code>false, true</code></td>
<td>Domain assignment depends on domain</td>
</tr>
<tr>
<td>id</td>
<td>1</td>
<td>Organization ID</td>
</tr>
<tr>
<td>members</td>
<td>#{array of user objects}</td>
<td>Array with complete Payload of the users being member of the organization</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Please see <a href="#">User</a> for more</td>
</tr>
<tr>
<td>name</td>
<td>Chrispresso Inc.</td>
<td>Organization name</td>
</tr>
<tr>
<td>note</td>
<td>Manufacturer of individual coffee products.</td>
<td>Note being available via web, console and API</td>
</tr>
<tr>
<td>shared</td>
<td><code>true, false</code></td>
<td>Defines if the organization is a sharing one</td>
</tr>
<tr>
<td>updated_at</td>
<td>2021-03-22T12:47:54.807Z</td>
<td>Last update time</td>
</tr>
<tr>
<td>updated_by</td>
<td>#{user object}</td>
<td>Complete Payload of the user that updated the organization</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Please see <a href="#">User</a> for more</td>
</tr>
<tr>
<td>updated_by_id</td>
<td>1</td>
<td>User ID that updated the organization</td>
</tr>
</tbody>
</table>

### Group

**Tip:** The following indice contains below mentioned information: *_.group*
### Table 7: Group-Index

<table>
<thead>
<tr>
<th>Field</th>
<th>Sample Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>active</td>
<td>true, false</td>
<td>Defines if group is active (available)</td>
</tr>
<tr>
<td>assignment_timeout</td>
<td>null, 30</td>
<td>Time in minutes an agent can be inactive until the ownership is removed</td>
</tr>
<tr>
<td>created_at</td>
<td>2021-03-24T23:55:06.980Z</td>
<td>Time stamp of group creation</td>
</tr>
<tr>
<td>created_by_id</td>
<td>1</td>
<td>User ID that created the group</td>
</tr>
<tr>
<td>email_address</td>
<td>Contains these attributes:</td>
<td>Contains all available information of the groups email address</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• active: true</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• channel_id: 3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• created_at: 2021-03-24T23:54:58.187Z</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• created_by_id: 3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• email: <a href="mailto:alias@domain.tld">alias@domain.tld</a></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• id: 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• note: null</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• realname: Zammad GmbH</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• updated_at: 2021-03-24T23:54:58.187Z</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• updated_by_id: 3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• preferences: null</td>
</tr>
<tr>
<td>email_address_id</td>
<td>3</td>
<td>ID of email address</td>
</tr>
<tr>
<td>follow_up_assignment</td>
<td>true, false</td>
<td>Defines if owners are still assigned after follow ups</td>
</tr>
<tr>
<td>follow_up_possible</td>
<td>yes, no</td>
<td>Defines if following up on a closed ticket is possible</td>
</tr>
<tr>
<td>id</td>
<td>1</td>
<td>Group ID</td>
</tr>
<tr>
<td>name</td>
<td>Users, Sales</td>
<td>Group name</td>
</tr>
<tr>
<td>note</td>
<td>null</td>
<td>Notes for the group available via web, console and API</td>
</tr>
<tr>
<td>signature</td>
<td>Contains these attributes:</td>
<td>Contains all available information of the groups signature</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• active: true</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• body: &lt;br&gt;#{user.firstname} &lt;br&gt;#{user.lastname}&lt;br&gt;&lt;br&gt;That Inc</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• created_at: 2021-03-03T14:50:19.775Z</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• created_by_id: 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• id: 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• name: default</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• note: null</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• updated_at: 2021-03-03T14:50:19.775Z</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• updated_by_id: 1</td>
</tr>
<tr>
<td>signature_id</td>
<td>1</td>
<td>Signature ID</td>
</tr>
<tr>
<td>updated_at</td>
<td>2021-03-24T23:55:06.980Z</td>
<td>Time stamp of last group update</td>
</tr>
<tr>
<td>updated_by_id</td>
<td>3</td>
<td>User ID that updated group</td>
</tr>
</tbody>
</table>
## CTI Log

**Tip:** The following indice contains below mentioned information: *cti_log*

### Table 8: CTI Log-Index

<table>
<thead>
<tr>
<th>Field</th>
<th>Sample Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>call_id</td>
<td>00006</td>
<td>Unique Call ID</td>
</tr>
<tr>
<td>comment</td>
<td>&quot;&quot;</td>
<td>Optional comment</td>
</tr>
<tr>
<td>created_at</td>
<td>2021-03-22T11:48:01.703Z</td>
<td>Creation date of Call</td>
</tr>
<tr>
<td>direction</td>
<td>in, out</td>
<td>Call direction</td>
</tr>
<tr>
<td>done</td>
<td>true, false</td>
<td>Defines if call displays as “to do” within UI</td>
</tr>
<tr>
<td>duration_talking_time</td>
<td>27</td>
<td>Call duration in seconds</td>
</tr>
<tr>
<td>duration_waiting_time</td>
<td>77</td>
<td>Duration in seconds</td>
</tr>
<tr>
<td>end_at</td>
<td>2021-03-25T08:49:40.647Z</td>
<td>Time stamp of call end</td>
</tr>
<tr>
<td>from</td>
<td>493055571600</td>
<td>Calling number</td>
</tr>
<tr>
<td>from_comment</td>
<td>null, John, Doe</td>
<td>Display name of calling number if applicable</td>
</tr>
<tr>
<td>from_pretty</td>
<td>+49 30 55571600</td>
<td>Pretty version of from</td>
</tr>
<tr>
<td>id</td>
<td>8</td>
<td>Internal ID of entry</td>
</tr>
<tr>
<td>initialized_at</td>
<td>2021-03-25T08:47:56.753Z</td>
<td>Time stamp of call initialization, usually matches created_at</td>
</tr>
<tr>
<td>preferences</td>
<td>(Array)</td>
<td>Contains internal information if required</td>
</tr>
<tr>
<td>queue</td>
<td>null, 491711234567890</td>
<td>Queue the call was answered</td>
</tr>
<tr>
<td>start_at</td>
<td>2021-03-25T08:49:13.050Z</td>
<td>Time stamp the call was answered</td>
</tr>
<tr>
<td>state</td>
<td>hangup, voicemail</td>
<td>Last state of call</td>
</tr>
<tr>
<td>to</td>
<td>491711234567890</td>
<td>Dialed number</td>
</tr>
<tr>
<td>to_comment</td>
<td>null, John, Doe</td>
<td>Display name of called number if applicable</td>
</tr>
<tr>
<td>to_pretty</td>
<td>+491711234567890</td>
<td>Pretty version of to</td>
</tr>
<tr>
<td>updated_at</td>
<td>2021-03-25T08:49:40.647Z</td>
<td>Last update of entry</td>
</tr>
</tbody>
</table>

## Chat Session

**Tip:** The following indice contains below mentioned information: *chat_session*
Table 9: Chat Session-Index

<table>
<thead>
<tr>
<th>Field</th>
<th>Sample Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>chat</td>
<td>Contains these attributes:</td>
<td>Contains various preferences of the chat topic in charge</td>
</tr>
<tr>
<td></td>
<td>• active: true</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• block_country: null</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• block_ip: null</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• created_at: 2021-03-03T14:50:22.607Z</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• created_by_id: 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• id: 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• max_queue: 5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• name: default</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• note: &quot;&quot;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• preferences: {}</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• public: false</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• updated_at: 2021-03-03T14:50:22.607Z</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• updated_by_id: 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• whitelisted_websites: null</td>
<td></td>
</tr>
<tr>
<td>chat_id</td>
<td>1</td>
<td>ID of Chat topic</td>
</tr>
<tr>
<td>created_at</td>
<td>2021-03-25T10:26:24.376Z</td>
<td>Time stamp of chat creation</td>
</tr>
<tr>
<td>created_by_id</td>
<td>null</td>
<td>User that created the chat, place holder, currently always null</td>
</tr>
<tr>
<td>id</td>
<td>1</td>
<td>ID of Chat Session</td>
</tr>
<tr>
<td>messages</td>
<td>(Array) - Array entries contain these attributes:</td>
<td>Array with all messages of chat</td>
</tr>
<tr>
<td></td>
<td>• chat_session_id: 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• content: Hello dear customer</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• created_at: 2021-03-25T10:26:35.977Z</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• created_by_id: null, 3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• id: 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• updated_at: 2021-03-25T10:26:35.977Z</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>null, John Doe</td>
<td>Name agent set for chat user, if applicable</td>
</tr>
<tr>
<td>preferences</td>
<td>Contains these attributes:</td>
<td>Various internal Meta data of the session_id</td>
</tr>
<tr>
<td></td>
<td>• dns_name: host.domain.tld</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• geo_ip: {}</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• participants: Array, [&quot;47118371175780&quot;, &quot;47118371850300&quot;]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• remote_ip: 192.168.2.19</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• url: <a href="https://zammad.com/en/company/contact">https://zammad.com/en/company/contact</a></td>
<td></td>
</tr>
<tr>
<td>session_id</td>
<td>92f2909631f1ad5ff4d5d1e046952be8</td>
<td>Unique identifier</td>
</tr>
<tr>
<td>state</td>
<td>closed</td>
<td>Current state of chat session</td>
</tr>
<tr>
<td>tags</td>
<td>(Array), [&quot;order&quot;]</td>
<td>Tags applied to Chat Session by agent, if applicable</td>
</tr>
</tbody>
</table>
Warning: We currently do not support Docker environments in productive use. If you run Zammad on docker, it is fine. But we just support the application!

Note: Docker Compose environments require deeper system know how. If you’re not too familiar with Docker and the way it works, you may want to stick with the package installation instead.

Docker is a container-based software framework for automating deployment of applications. Compose is a tool for defining and running multi-container Docker applications.

This repo is meant to be the starting point for somebody who likes to use dockerized multi-container Zammad in production. The Zammad Docker image uses the stable branch of Zammad’s Git repo.

The Docker images are hosted on Dockerhub.

Warning: Never use the latest tag. Use a tag which has a version attached.

You need at least 4 GB of RAM to run the containers.

7.1 Install Docker Environment

This documentation expects you already have a working Docker Compose environment. You can find the required documentations for these steps below:

- Docker Engine
- Docker Compose
7.2 Getting started with zammad-docker-compose

7.2.1 Docker Compose Environment Variables

Zammad’s Docker Compose supports several environment variables that are not set by default. The best way to provide these is within the file `.env`.

In case our default `docker-compose.yml` is not good enough, please use `docker-compose.override.yml` to provide own changes.

**Docker Compose specific**

**RESTART:** `always` By default containers will be restarted in case they stopped for whatever reason.

**VERSION** This variables contains the version tag. Example: `3.6.0-20`

We update this string from time to time, Docker Hub may contain more current tags to the moment you’re pulling.

**Zammad**

**AUTOWIZARD_JSON:** `''` This variable allows you to provide initial configuration data for your instance. Autowizard JSON is out of scope of this documentation, however this example file should help you.

**RAILS_TRUSTED_PROXIES:** `['127.0.0.1', '::1']` By default Zammad trusts localhost proxies only.

Only change this option if you know what you’re doing!

**Elasticsearch**

**ELASTICSEARCH_ENABLED:** `true` Setting this variable to false will allow you to run your Zammad without Elasticsearch. Please note that we strongly advise against doing so.

**ELASTICSEARCH_HOST:** `zammad-elasticsearch` Provide a host name or address to your external Elasticsearch cluster.

**ELASTICSEARCH_PORT:** `9200` Provide a different port for Elasticsearch if needed.

**ELASTICSEARCH_SCHEMA:** `http` By default Elasticsearch is reachable via HTTP.

**ELASTICSEARCH_NAMESPACE:** `zammad` With this name space all Zammad related indexes will be created. Change this if you're using external clusters.

**ELASTICSEARCH_REINDEX:** `true` By default the docker-compose will always re-index upon a restart. On big installations this may be troublesome.

| Warning: | Disabling this setting requires you to re-index your search index manually whenever that’s needed by upgrading to a new Zammad version! |

**ELASTICSEARCH_SSL_VERIFY:** `true` Allows you to let the compose scripts ignore self signed SSL certificates for your Elasticsearch installation if needed.
Memcached

MEMCACHE_SERVERS: **zammad-memcached:11211** Provide your own Memcached instance if you already have one existing.

**Warning:** Was MEMCACHED_HOST before 5.0.x!

Redis

REDIS_URL: `redis://zammad-redis:6379` Provide your own Redis instance if you already have one.

**Warning:** This method currently does not allow authentication.

Nginx

NGINX_PORT: **8080** The port Nginx will listen on.

NGINX_SERVER_NAME: _ By default the Nginx container of Zammad will respond to all request. You can provide your IP / FQDN if you want to.

NGINX_SERVER_SCHEME: `\$scheme` If the Nginx container for Zammad is not the upstream server (aka you’re using another proxy in front of nginx) $scheme may be wrong. You can set the correct scheme http or https if needed.

**Tip:** Can't login because of CSRF token errors?
This usually affects systems with more than one proxy server only. For this to function you may have to tell your web server directly which connection type was used.

**Warning:** Do not use below options if you're unsure, they may technically be a security issue!

The following options expect HTTPS connections which should be your goal.

nginx

Within your virtual host configuration, locate both directives `proxy_set_header X-Forwarded-Proto` and replace $scheme by https.

Within your virtual host configuration just above the first ProxyPass directive insert:

```bash
RequestHeader set X_FORWARDED_PROTO 'https'
RequestHeader set X-Forwarded-Ssl on
```

ZAMMAD_RAILSSERVER_HOST: **zammad-railsserver** Host name of the rails server container.

ZAMMAD_RAILSSERVER_PORT: **3000** Port of Zammads rails server.

Please also note **Configuration via Environment Variables** in this regard.

ZAMMAD_WEBSOCKET_HOST: **zammad-websocket** Host name of Zammads websocket server.

7.2. Getting started with zammad-docker-compose
Zammad

ZAMMAD_WEBSOCKET_PORT: 6042  Port of Zammad's websocket server.
   Please also note Configuration via Environment Variables in this regard.

**PostgreSQL**

POSTGRESQL_HOST: zammad-postgresql  Host name of your PostgreSQL server. Use your own if you already have one.

POSTGRESQL_PORT: 5432  Adjust the Port of your PostgreSQL server.

POSTGRESQL_USER: zammad  The database user for Zammad.

POSTGRESQL_PASS: zammad  The password of Zammad's database user.

POSTGRESQL_DB: zammad_production  Zammad's database to use.

POSTGRESQL_DB_CREATE: true  By default we will create the required database.

   Note:  On own database servers this setting might be troublesome.

RSYNC_ADDITIONAL_PARAMS: --no-perms --no-owner  By default the compose will copy data without permissions and owners. This may not fit for your storage driver.

7.2.2 Step 1: Clone GitHub repo

$ git clone https://github.com/zammad/zammad-docker-compose.git
$ cd zammad-docker-compose

7.2.3 Step 2: Setting vm.max_map_count for Elasticsearch

$ sysctl -w vm.max_map_count=262144

Tip:  Mac OS users please also have a look on Issue 27

7.2.4 Step 3: Adjust Environment as needed

In some cases our default environment is not what a docker-compose user is looking for. To remove complexity from this page, we outsourced information on this topic.

See Docker Compose Environment Variables
7.2.5 Step 4: Start Zammad using DockerHub images

**Warning:** Before starting your containers ensure to not use default login data for your Zammad database! See Step 3!

```bash
$ docker-compose up
```

7.3 Next steps

With this Zammad technically is ready to go. However, you’ll need to follow the following further steps to access Zammad’s Web-UI and getting started with it.

1. *Connect Zammad with Elasticsearch*
2. *Configure the webserver*
3. *First steps*
4. You may also find Zammad’s *Console* commands useful

If you expect usage with 5 agents or more you may also want to consider the following pages.

- *Performance Tuning*
- *Configure Database server*
Warning: We currently do not support Kubernetes installations in productive use.

Kubernetes (k8s) is an open-source system for automating deployment, scaling, and management of containerized applications.

Helm is the package manager for Kubernetes.

This repo is meant to be the starting point for somebody who likes to use dockerized multi-container Zammad on Kubernetes. The Zammad Docker image uses the stable branch of Zammad’s Git repo.

The used Docker images are hosted on Dockerhub.

You need the Helm binary installed / initialized and at least 4 GB of free RAM in the Kubernetes cluster run the containers.

# Add Helm repo
$ helm repo add zammad https://zammad.github.io/zammad-helm

# Install / Upgrade Zammad
$ helm upgrade --install zammad zammad/zammad --namespace=zammad


CHAPTER

NINE

INSTALLATION ON UNIVENTION CORPORATE SERVER VIA APP CENTER (EOL)

Danger: WARNING: END OF LIFE

Please note that Zammad GmbH announced the end of life of the Zammad-App for Univention! Users where already alarmed by Email late 2021 about this upcoming change.

We’re providing an export guide, please see: Export Zammad on Univention for migration.

If you’re planning to install Zammad on Univention right now, please consider using either package, docker compose, or source installations as alternative.

Note: As Zammad is using Docker Compose for Univention Corporate Server, the minimum requirement is UCS 4.3-2 errata 345.

Univention Corporate Server (UCS) is an enterprise server with focus on identity and infrastructure management. With its marketplace called App Center it can easily be extended by solutions like Zammad that benefit from integrations with the LDAP directory service and the mail infrastructure.

Click here to learn more about Univention and what it can do for you.

9.1 Prerequisites

To install the Zammad app on UCS, please ensure that you’re using at least UCS 4.3-2 errata 345. The basic installation will already meet our requirement. You’ll need the following additional things:

• An email server (no matter if handled via UCS or with an external system) for notifications, as you can’t use sendmail in our Docker setup!

• You should at least have 2 CPU-Cores and 4GB of free RAM.

Note: Running the Zammad app with less than 4GB free RAM will lead to unexpected errors!

You see, that’s not much - so go a head with the installation.
9.2 Installing Zammad

Zammad for Univention is end of life. For this reason you can no longer install the app. Below parts are provided for historic reasons.

9.2.1 Values we automatically change during the UCS-Setup

In order to make the installation as complete and convenient as possible, we're changing the following default values to the following:

<table>
<thead>
<tr>
<th>value</th>
<th>default value</th>
<th>new value</th>
</tr>
</thead>
<tbody>
<tr>
<td>notification sender</td>
<td>Notification Master <code>&lt;noreply@#{config.fqdn}&gt;</code></td>
<td>Zammad <code>&lt;noreply@[FQDN-of-UCS]&gt;</code></td>
</tr>
<tr>
<td>maximum email size</td>
<td>10 MB</td>
<td>35 MB</td>
</tr>
<tr>
<td>FQDN</td>
<td><code>{FQDN-of-UCS}</code></td>
<td><code>{FQDN-of-UCS}:10412</code></td>
</tr>
<tr>
<td>HTTP-Type</td>
<td><code>&lt;empty&gt;</code></td>
<td>https</td>
</tr>
<tr>
<td>Allow customer registration</td>
<td>true</td>
<td>false</td>
</tr>
<tr>
<td>LDAP configuration</td>
<td><code>&lt;empty&gt;</code></td>
<td>Full LDAP-Configuration prepared</td>
</tr>
<tr>
<td>LDAP activated</td>
<td><code>&lt;empty&gt;</code></td>
<td>false</td>
</tr>
</tbody>
</table>

Note: Please note that the Zammad-LDAP integration is pre filled with authentication data and the group mapping Zammad-Admin to the Admin-Role and Zammad-Agent to the Agent-Role. You can use those security groups. LDAP synchronization is disabled during installation, as activating it would disable the installation wizard of Zammad, which is needed to setup your Zammad instance properly.

Note: Please note that these settings are updated automatically, if you update FQDN and Port settings within the Univention App-Settings.

9.3 First steps you should consider

The most important part is obvious: Run the wizard and insert the information for your admin account.

Warning: If the email address is used within UCS, you need to ensure that your user account within UCS has the needed Admin-Group, as otherwise a LDAP synchronization will downgrade your user account to the setup role!

You can now enter your company name and upload a company logo, if you want to. (the company name is mandatory). The system URL has been set by our installation routine already, you should be good to continue without changing it.

Note: Changing the system URL might lead to broken links within notification Emails.
For the notification sender, you should use SMTP, as the Docker container does not come with any sendmail or local MTA. If you choose local MTA, Zammad will not be able to send you any notifications.

The last step offers you to add your first email accounts to Zammad. You’re free to skip this step, you can configure your accounts later, as well.

Zammad is now ready to go.

The identity management integration with UCS LDAP directory allows the system administrator to maintain the users at one single point. If you want to take advantage of UCS identity management integration, you need to do the following before hand:

- Add your desired Zammad admin users to the user group **Zammad-Admin**.
- Add your desired Zammad agents to the user group **Zammad-Agent**.
- All user accounts that are not covered by the default group mapping, will be added in the Zammad customer role.

You can now go to Admin-Settings -> Integration -> LDAP and simply activate LDAP. The first LDAP synchronization will start shortly thereafter - Zammad will then synchronize user account data with the UCS LDAP directory hourly.

**Note:** You’re free to change the group-role mapping at any time. See Configuring LDAP integration for more information.

### 9.3.1 Further configuration

The rest of the configuration is pretty straight forward and applies to our default. We split our documentation into two further parts that will be of your interest:

- **Admin documentation:** this documentation holds any information about how to configure Zammad via WebApp.
- **User documentation:** this documentation holds a complete user documentation (how to work with Zammad).

### 9.4 Further information

The following sub pages might come in handy and help you to understand how the app works or on how to solve an issue.

#### 9.4.1 Export Zammad on Univention for migration

This page will guide you through the process on how to export your instance data on Univention systems. The provided backup suits all supported platforms for a regular backup restore. If you require help, please contact Zammad-Sales for a discounted migration workshop.

The provided method is provided as is and does not receive any community support.

**Warning:** Our export has some limitations you have to be aware of:

- **Only one running Zammad instance per host is supported.**
  - If you might have several instances running on the same host, stop all instances except the one you’re exporting.
Zammad

- Zammad for UCS version 5.1.1-16 only is supported. Using older versions will fail, the backup method *is not* supported!
- The export script expects root permissions on the host in question.
- The script ensures there’s no delayed jobs left and stops parts of the stack. *Expect a downtime.*
- Due to limitations on Univention, the update scripts for 5.1 are required to download an extra script from vps1337.dc.zammad.com.
  
  For this reason, please ensure that above host can be connected via https during upgrade time.

**Hint:**

**About our setup:**

We’ve verified our scripts function on both, Univention 4.4 and Univention 5.0. Both test scenarios consisted on a clean 5.1.1-16 installation and one upgrade installation from 4.1.0-52.

2 Clean installations are not available for normal UCS users as the App has been set to EOL.

---

**Upgrade your Zammad app**

Before you can run the export, please ensure to upgrade your Zammad installation to 5.1.1-16. The update should run as smooth as always.

Before the actual upgrade, our pre-installation script will download an extra export script that is required for the actual migration at an later point. If the can’t retrieve the script successful, the upgrade process will abort.

**Export your Zammad data for the migration**

**Warning:** Important

Ensure that your app has version 5.1.1-16 - if that’s not the case, upgrade your app.

During the upgrade we’ve placed an export script on your system for you. This script is located on the host that’s running the Zammad app.

**Danger:** Downtime ahead

Make sure to run the migration during a planned maintenance window.

**Tip:** To reduce maintenance you may wanna install Zammad before hand. The easiest installation type is the package installation.

If you’re ready to go, login as to your Univention hosts SSH console as administrative user (administrator on default systems).

After logging in, ensure to have an evalated session by running sudo -s. The export script is located in /var/lib/univention-appcenter/apps/zammad/.

---

62 Chapter 9. Installation on Univention Corporate Server via App Center (EOL)
Executing `/var/lib/univention-appcenter/apps/zammad/ucs-export.sh` will start the backup process. The process will ensure that there’s no delayed jobs waiting for completion (mandatory) and also enable maintenance mode for you.

The backup is being created without any further warnings - this process stops Zammad containers and does not start them again.

---

**Note: Sample output**

To give you an idea what a successful export looks like, you can find a console output below. IDs and names may differ.

```
# ============== #
# Start of Zammad UCS export ... #
# ============== #
# ... Ensuring permissions ... #
# ... Ensuring limits ... #
# ... Getting container IDs #
# ... Enabling maintenance mode #
# ... Ensuring no delayed jobs #
# ... Checking delayed jobs (0) #
# Listing 'docker ps' before ... #
CONTAINER ID   IMAGE                     COMMAND                               CREATED                STATUS              PORTS                   NAMES
--------------  ---------------------------------------  ------------------------  ---------------------  ---------------------  -----------------------  --------
d8d9ff288b95   zammad/zammad-docker-compose:zammad-5.1.1-16   
               
               
               
               
               
               

# ... Stopping Zammad containers #
```

(continues on next page)
You can safely ignore this warning:

... WARNING: You don't seem to have any attachments in the file system!
... Please consult https://docs.zammad.org/en/latest/appendix/backup-and-
---restore/troubleshooting.html
... Creating empty storage directory so the backup can continue ...
After exporting

Please ensure to check the provided console output for errors. If there were no errors, please continue with the restoration process.

You can find the relevant backup files in /var/lib/univention-appcenter/apps/zammad/data/backup/.

If the migration was a success, stop and remove the Univention app.

**Hint:** Please note that removing the app does also remove the LDAP users created during the installation. If you use our default LDAP mapping you may have to adjust it before hand!

9.4.2 Univention App-Settings

**Danger:** WARNING: END OF LIFE

Please note that Zammad GmbH announced the end of life of the Zammad-App for Univention! Users where already alarmed by Email late 2021 about this upcoming change.

We’re providing an export guide, please see: Export Zammad on Univention for migration.

If you’re planning to install Zammad on Univention right now, please consider using either package, docker compose, or source installations as alternative.

**Note:** The App-Settings part is only valid starting with App version 3.1.0-7.

Within the management interface of Univention, you can change some access related settings:

- FQDN of Zammad (Domain you can access it from)
- Another Port
- Other certificates beside the Univention certificate

**Note:** Some settings require you to have a combination of the above settings. If the combination of settings is not met, the update script will automatically revert these changes.

This ensures that your Univention Host stays operational. Please do not try to trick the scripts, it might cause outages.

If you’re using the default settings, we’ll empty the values of the text fields to reduce confusion. This is not a bug, but a “feature”.

9.4. Further information
Zammad

Using another FQDN

This consists of two settings: a selection and a text field. The default setting of the App is Default UCS-Hostname which will use the FQDN of the Univention-Host (and it's SSL-Certificate).

Note: In order to use custom hostnames, please also ensure to use a custom certificate.

If you set Custom Hostname, you'll need to enter a hostname in the text field below. Ensure that your Univention host can resolve the hostname (and it’s pointing to the host in question!).

Note: We won’t create any DNS entries or certificates during this process.

Using another Port

Currently you can choose between Default Highport and Port 8443. By default, we'll use Port 10412 for Zammad, if you decide for Port 8443, we’ll handle the firewall steps needed and adjust the vHosts-Port.

Warning: Please ensure that if you choose another Port, that it’s not used already! We do not verify this!

Note: Please note that for technical reasons (how Univention and Zammad work) it’s not possible to use Zammad on Port 443 or within a sub directory.

Using other certificates

By default we’re using the Univention-Host certificate (Univention (default)). In some cases (and especially if you’re using custom hostnames) this might be troublesome.

Note: You can use custom certificates without changing the hostname.

Warning: We’re not verifying if the certificates are valid in any way (e.g. still valid in time and if the hostname is inside). This step might follow, but please be aware that this might lead to certificate issues.

If you choose Let’s Encrypt, please ensure that you already have installed the Let’s Encrypt App (by Univention GmbH) and also already acquired a certificate via it. If you’re applying the settings, we’ll check for the following two files:

```shell
/etc/univention/letsencrypt/signed_chain.crt
/etc/univention/letsencrypt/domain.key
```

If we can't find these, we’ll revert to the default Univention certificate.

You can also choose to use your very own certificate by selecting Custom Certificate. For this it's important to know, that we expect the certificate to be within a specific location (/etc/univention/ssl/). Within the two text fields, you'll need to provide the filenames of your certificate and your certificate-key.
These certificates can be kept in a sub folder. If we cannot find either of the two files, we reset the setting to the default Univention certificate.

### 9.4.3 Running console commands on an Univention-Host

**Danger:** **WARNING: END OF LIFE**

Please note that Zammad GmbH announced the end of life of the Zammad-App for Univention! Users where already alarmed by Email late 2021 about this upcoming change.

We’re providing an export guide, please see: *Export Zammad on Univention for migration*.

If you’re planning to install Zammad on Univention right now, please consider using either *package*, *docker compose*, or *source* installations as alternative.

In some cases you might need to access Zammads rails console on the Univention-Host. For this, you’ll need to get the correct container ID first.

Univention will hold this information for you, you can get it like that:

```
$ ucr get appcenter/apps/zammad/container
```

Now where we have our ID, you can run any command from the *Console* section with either:

```
$ docker exec -i "{Container-ID}" rails r "{COMMAND}"
```

or -if you need a console for more commands- by:

```
$ docker exec -i "{Container-ID}" rails c
```

**Note:** Please replace `{Container-ID}` in the above commands by the ID the first command returns. Replace `{COMMAND}` by any rails command Zammad supports.

That’s it!

### 9.4.4 Issues you might encounter

**Danger:** **WARNING: END OF LIFE**

Please note that Zammad GmbH announced the end of life of the Zammad-App for Univention! Users where already alarmed by Email late 2021 about this upcoming change.

We’re providing an export guide, please see: *Export Zammad on Univention for migration*.

If you’re planning to install Zammad on Univention right now, please consider using either *package*, *docker compose*, or *source* installations as alternative.

Below we have gathered information to problems that might occur in combination with Univention.
Zammad can’t communicate with external systems

In rare cases (sometimes even right after installation), Zammad won’t be able to communicate with e.g. external Email servers. Simply restart the Zammad app in the App Center module in the UCS management system and it should be enough to get it back working.

**Warning:** Never change any configurations the Zammad-App scripts create and work with! This will lead to unexpected issues and loss of configurations upon update!
Note:

**Better safe than sorry** Before updating to a new version, please have a look into the release notes. These will provide further information on new feature and fixes, but also technical remarks that may be relevant during an upgrade!

**What about Zammad upgrade paths...?** In general we do not encourage you to skip Zammad versions or have long update cycles. Zammad potentially stores very sensitive information (personal information) which is why updating is very important.

If you don’t have time for updating all the time (nobody got time for that, right?), please consider using Zammad hosting for your and your customers safety.

In case you couldn’t update for a longer time, please ensure to at least update from **mayor to mayor** version. Big version jumps *may* work but usually go terribly wrong. As example, expecting the current stable version of Zammad being 5.1 and your instance being on Zammad 2.4, your path would look like so: 2.4 → 3.0 → 4.0 → 5.0 → latest stable (5.1)

---

### Package

### Source

### Docker Compose

**Step 1: Ensure dependencies** Before proceeding, double-check that your system environment matches [Zammad's requirements.](#)

**Step 2: Stop Zammad**

```bash
$ systemctl stop zammad
```

**Step 3: Backup Zammad** See [Backup and Restore](#) for more information.

**Step 4: Clear Zammad cache**

```bash
$ zammad run rails r "Cache.clear"
```

**Step 5: Update Zammad** Ubuntu / Debian

- CentOS
- OpenSUSE / SLES

```bash
$ apt update
$ apt upgrade
```
$ yum update zammad

$ zypper ref
$ zypper up

**Warning:** The package comes with maintenance scripts that will run regular tasks during updates for you. However

Do not run Zammad updates unattended and **always** have a look on the outputs these helper scripts generate. Ignoring said output may lead to incomplete updates that may corrupt data or lead to issues you find **way too late**.

Step 6: **Run required extra steps** Extra steps needed for updates are mentioned in our release news. *Updating Elasticsearch* may be relevant in this step.

Step 7: **Log into Zammad** Yes, that’s it!

**Danger:** Zammad’s former `scheduler.rb` script has changed and is now called `background-worker.rb`. Please ensure to reinstall the service - see *Step 6: Start Zammad or install as service*!

Step 1: **Ensure dependencies** Before proceeding, double-check that your system environment matches Zammad’s requirements.

**Tip:** Ruby version changed? Please see *Installation part of source code installation*

Step 2: **Stop Zammad and Clear Zammad cache** Before you continue, stop your Zammad processes.

```
$ rails r "Cache.clear"
```

Step 3: **Download Zammad to your system** Get the latest stable release of Zammad here, or find the initial version at https://ftp.zammad.com.

```
$ cd /opt
$ wget https://github.com/zammad/zammad/archive/stable.tar.gz
$ tar -xzf stable.tar.gz --strip-components 1 -C zammad
$ chown -R zammad:zammad zammad/
$ rm -f stable.tar.gz
```

For security reasons, ensure that your database configuration is readable for the Zammad user only.

```
$ chmod 600 /opt/zammad/config/database.yml
$ chown zammad:zammad /opt/zammad/config/database.yml
```

Step 4: **Install Gems**
$ su - zammad
$ cd /opt/zammad
$ gem install bundler

**PostgreSQL**

**MySQL / MariaDB**

$ bundle install --without test development mysql

$ bundle install --without test development postgres

**Step 5: Stop Zammad services**  Stop the application server, websocket server and scheduler.

**Step 6: Upgrade your database**

$ su - zammad
$ rake db:migrate
$ rake assets:precompile

**Step 7: Synchronize Zammad’s translation files**

$ su - zammad  # ignore if you haven’t exited the Zammad user
$ rails r "Locale.sync"
$ rails r "Translation.sync"

**Step 8: Start Zammad services**  Start the application server, web socket server and scheduler.

**Step 9: Log into Zammad**  Yes, that’s it!

**Warning:** Updates may require extra steps or introduce breaking changes.

Always check the upgrade notes first.

**Note:** Incomplete documentation

Sorry, but this documentation part is outdated. We will rework this part later, but can’t tell when yet.

Please feel welcome to provide a pull request if you find spare time!

$ docker-compose stop
$ git pull
$ docker-compose pull
$ docker-compose up

Start Zammad building Docker images locally with development branch

- GIT_BRANCH=develop docker-compose -f docker-compose-build.yml up

Recreate locally built images

- GIT_BRANCH=develop docker-compose -f docker-compose-build.yml build –no-cache

Open shell in running Zammad image

- docker-compose exec zammad /bin/bash
Port compatibility error

- The nginx container may have compatibility problems with other machines or services pointing to port 0.0.0.0:80.
  So to fix this, we’ll just have to modify the file `docker-compose.override.yml` and select different ports

### 10.1 Updating Elasticsearch

**Warning:** Updating Elasticsearch does not automatically update its plugins! This usually isn’t an issue if Zammad is being updated right after Elasticsearch.

If you want to upgrade your elasticsearch installation, please take a look at the [elasticsearch documentation](https://www.elastic.co/guide/en/elasticsearch/reference/current/) as it will have the most current information for you.

If, for whatever reason, you need to rebuild your search index after upgrading, use:

```
$ zammad run rake zammad:searchindex:rebuild
```

**Hint:** Zammad 5.2 comes with changes

As of Zammad 5.2 the reindex command has changed! You will still be able to use the old method until Zammad 6, however, will receive a deprecation warning.

**Warning:** This step may fail if Zammad is under heavy load: Elasticsearch locks the indices from deletion if you’re pumping in new data, like receiving a new ticket. (This only applies to single-node deployments, not clusters.)

If it does, try killing Zammad first:

```
$ systemctl stop zammad
$ zammad run rake zammad:searchindex:rebuild
$ systemctl start zammad
```

CONFIGURE THE WEBSERVER

You can find current sample configuration files for your webserver within contrib/ of your Zammad installation.

If you’re using the package installation, Zammad attempts to automatically install a configuration file to your nginx for you.

**Note:** The Zammad installation will not automatically set any host- or server name for you.

Docker Compose / Kubernetes users
Please also note the environment information on this page

11.1 Get a ssl certificate (recommended)

Don’t know how to get SSL certificates and install them on a webserver yet? The guide within the tabs below can help you jumping in.

I don’t need that

letsencrypt

public, paid CA

self-signed (discouraged)

You either already know what you’re doing, you’re developing or like the danger.

letsencrypt is an easy and free way to retrieve valid ssl certificates. These certificates are valid for 90 days and can be renewed automatically.

The two most common tools are certbot and acme.sh.

certbot

acme.sh

**Hint:** If not happened automatically, you may need to install the nginx or apache plugin for certbot:

```
python3-certbot-nginx OR python3-certbot-apache
```

During the first certbot run it will request additional information once. Replace `<webserver>` in below command by either apache, httpd or nginx and to match your setup.
Certbot will now attempt to issue a certificate for you. If successful, certbot will ask you if you want to [1] not redirect or [2] redirect automatically. You can choose to not redirect if you plan to use the sample configuration of Zammad. If not, select [2] redirect.

From this moment on, certbot will automatically renew your installed certificates if they’re valid for another 30 days or less.

**Hint:** Not exactly what you’re looking for?

The **certbot documentation** has a lot more use cases than we cover here.

---

**Hint:** The most reliable way is to use the standalone method.

First of all you’ll need to issue your certificate. **acme.sh** will save this certificate to `/root/.acme.sh/ <your-domain>/`

```
acme.sh --issue --standalone -d zammad.example.com
```

It’s not recommended to use the just stored certificates directly. Instead you should install the certificate to a directory of your choice.

We’re using `/etc/ssl/private/` in this case, but you can use any directory you like.

**Warning:** Ensure to adjust value for `--reloadcmd` as this will ensure that `acme.sh` reloads your webserver automatically after getting a renewal. Replace `<webserver>` by either `apache2`, `httpd` or `nginx`

```
acme.sh --install-cert -d zammad.example.com \
  --cert-file /etc/ssl/private/zammad.example.com.pem \
  --key-file /etc/ssl/private/zammad.example.com.key \
  --fullchain-file /etc/ssl/private/zammad.example.com.full.pem \
  --reloadcmd "systemctl force-reload <webserver>"
```

From this moment on, `acme.sh` will automatically renew your installed certificates if they’re valid for another 30 days or less.

**Hint:** Not exactly what you’re looking for?

The **acme.sh documentation** has a lot more use cases than we cover here.

---

If you prefer to use certificates from other official CAs than let’s encrypt, you can do so as well. Just get your certificate bundle from the source you prefer and continue with **Adjusting the webserver configuration.**

---

**Note:** I’m new to SSL certificates. Where can I get a certificate?

The easiest way to get certificates is to buy an annual subscription through a commercial CA, such as:

- Sectigo (formerly Comodo)
Another way is to use self signed certificates from your own CA. In general you shouldn’t use this option when you have users accessing Zammad that can’t verify your certificates.

Beside creating own certificates via e.g. XCA or Microsoft CA, you can also generate a certificate really quick like so:

On any system with openssl installed, you can run below command. Provide the requested information and ensure to provide the fqdn of Zammad when being asked for Common Name (e.g. server FQDN or YOUR name).

```
openssl req -newkey rsa:4096 -nodes -x509 -days 1825
-keyout key.pem -out certificate.pem
```

Above command creates a certificate that’s valid for 5 years. It will write the certificate and private key to the current directory you’re in. If you want to check your certificate you just created, you can use the following command.

```
openssl x509 -text -noout -in certificate.pem
```

**Hint:** Not good enough for you?

If above command is not good enough for you, the OpenSSL documentation is a good place to learn more.

---

### 11.2 Adjusting the webserver configuration

**Warning:** For a quick start, we’re installing a HTTP configuration. You should **never** use HTTP connections for authentication - instead, we encourage you to use HTTPS!

If Zammad scripts automatically installed your webserver configuration file, ensure to not rename it. Below we’ll cover HTTPs for above reason.

nginx (default)
apache2
local testing or other proxy servers

**Step 1 - Get a current config file** Copy & overwrite the default `zammad.conf` by using

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

**Note:**
Your nginx directories may differ, please adjust your commands if needed.
Most common:

- `/etc/nginx/conf.d/`
- `/etc/nginx/vhosts.d/`
- `/etc/nginx/sites-enabled/`
**Step 2 - Adjust the config file** Adjust the just copied file with a text editor of your choice (e.g. vi or nano).

Locate any `server_name` directive and adjust `example.com` to the subdomain you have chosen for your Zammad instance.

Now you’ll need to adjust the path and file names for your ssl certificates your obtained on the prior steps. Adjust the following directives to match your setup:

- `ssl_certificate` (your ssl certificate)
- `ssl_certificate_key` (the certificates private key)
- `ssl_trusted_certificate` (the public CA certificate)

**Note:** Technically this is not a hard requirement, but recommended!

**Hint:** Don’t have a dhparam.pem file yet?

You can easily adapt below example to generate this file. It will improve HTTPs security and thus should be used.

You can find the path by looking at your webserver configuration by looking for:

- `ssl_dhparam` directive (nginx)
- `SSLOpenSSLConfCmd DHParameters` directive (apache2)

```bash
$ openssl dhparam -out <path>/dhparam.pem 4096
```

(Optional) - Adjust HTTPs configuration Our default configuration aims for a broad support of enduser devices. This may not fit your needs - Mozilla has a great ssl-config generator that should help you to meet your requirements!

**Step 3 - Save & reload** Reload your nginx `systemctl reload nginx` to apply your configuration changes.

**Step 1 - Ensure required modules are enabled** Zammad requires modules that are not enabled by default. By default use `a2enmod` (not CentOS) to do so.

```bash
a2enmod
via configuration file (CentOS)
```

```bash
$ a2enmod proxy proxy_html proxy_http proxy_wstunnel headers ssl
$ systemctl restart apache2
```

add/uncomment the appropriate `LoadModule` statements in your Apache config:

```bash
# /etc/httpd/conf/httpd.conf

LoadModule headers_module modules/mod_headers.so
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_html_module modules/mod_proxy_html.so
LoadModule proxy_http_module modules/mod_proxy_http.so
LoadModule proxy_wstunnel_module modules/mod_proxy_wstunnel.so
```

Don’t forget to restart your apache.
Step 2 - Get a current config file

Note: Package installations attempt to copy a `zammad.conf` to your webservers configuration directory. Do not rename this file!

Copy & overwrite the default `zammad.conf` by using

```
$ cp /opt/zammad/contrib/apache2/zammad_ssl.conf /etc/apache2/sites-enabled/zammad.conf
```

Note:
Your apache directories may differ, please adjust your commands if needed.
Most common:
- `/etc/apache2/conf.d/`
- `/etc/httpd/vhosts.d/`
- `/etc/apache2/sites-available/`

Step 3 - Adjust the config file

Adjust the just copied file with a text editor of your choice (e.g. `vi` or `nano`).

Locate any `ServerName` directive and adjust `example.com` to the subdomain you have chosen for your Zammad instance.

Now you’ll need to adjust the path and file names for your ssl certificates you obtained on the prior steps. Adjust the following directives to match your setup:
- `SSLCertificateFile` (your ssl certificate)
- `SSLCertificateKeyFile` (the certificates private key)
- `SSLCertificateChainFile` (the public CA certificate)

Note: Technically this is not a hard requirement, but recommended!

Hint: Don’t have a `dhparam.pem` file yet?  

You can easily adapt below example to generate this file. It will improve HTTPs security and thus should be used.

You can find the path by looking at your webserver configuration by looking for:
- `ssl_dhparam` directive (nginx)
- `SSLOpenSSLConfCmd DHParameters` directive (apache2)

```
$ openssl dhparam -out <path>/dhparam.pem 4096
```

(Optional) - Adjust HTTPS configuration

Our default configuration aims for a broad support of enduser devices. This may not fit your needs - Mozilla has a great `ssl-config generator` that should help you to meet your requirements!
Zammad

(Optional) - Enable the site

**Hint:** This step mostly depends on your selected folders and most often only affects sites-available folders.

Ubuntu / Debian / openSUSE

CentOS

```
$ a2ensite zammad
```

```
$ ln -s /etc/httpd/sites-available/zammad_ssl.conf /etc/httpd/sites-enabled/
```

Also, make sure the following line is present in your Apache configuration:

```
# /etc/apache2/apache2.conf (Ubuntu, Debian, & openSUSE)
# /etc/httpd/conf/httpd.conf (CentOS)
IncludeOptional sites-enabled/*.conf
```

**Step 4 - Save & reload** Reload your apache systemctl reload apache2 to apply your configuration changes.

Want to test locally first or use a different Proxy we don't support? The main application (rails server) is listening on http://127.0.0.1:3000.

If you’re using a proxy server, also ensure that you proxy the websockets as well. The websocket server listens on ws://127.0.0.1:6042.

**Tip:** If above ports are used by other applications already you may want to have a look at network options on our environment page.

**Warning:** Do not expose Zammad directly to the internet, as Zammad only provides HTTP!

If you just installed Zammad, you’ll be greeted by our getting started wizard. You now can continue with First steps.

**Hint:** You’re not seeing Zammads page but a default landing page of your OS?

Ensure that you did restart your webserver - also check if 000-default.conf or default.conf in your vhost directory possibly overrules your configuration.

Sometimes this is also a DNS resolving issue.

**Tip:** Can’t login because of CSRF token errors?

This usually affects systems with more than one proxy server only. For this to function you may have to tell your web server directly which connection type was used.

**Warning:** Do not use below options if you’re unsure, they may technically be a security issue!
The following options expect HTTPS connections which should be your goal.

nginx

apache2

Within your virtual host configuration, locate both directives `proxy_set_header X-Forwarded-Proto` and replace `$scheme` by `https`.

Within your virtual host configuration just above the first `ProxyPass` directive insert:

```
RequestHeader set X_FORWARDED_PROTO 'https'
RequestHeader set X-Forwarded-Ssl on
```
After successfully installing Zammad you’ll have a couple of options.

- *Migrate from a supported source*
- *Restore Zammad from an existing backup*
- Start from scratch (move on to the next section)

### 12.1 Getting Started Wizard

If you visit Zammad’s web page the first time, you’ll be greeted by its Getting Started Wizard. It will guide you through the first most important things.

**Step 1:** Create your very first administrator account  The fields should be fairly self explaining.

**Note:** Zammad does require the following password security by default:

- Password length of 10 or more
- 2 upper and 2 lower characters
- contains at least one digit

**Step 2:** Provide company information  You can upload a custom logo fitting to your company here. The instance address is detected automatically and only required adjustment in case it’s detected wrong.

All of these settings can be changed within Branding and System settings.

**Step 3:** E-Mail notification channel  By default Zammad uses sendmail - if that doesn’t fit you can change it to SMTP here.

**Warning:** Zammad uses noreply@<your-fqdn> as sender address by default. SMTP setups might fail - you may want to skip this step with choosing sendmail at this point. You can adjust it later!

**Step 4:** Your first email channel (optional)  If you want to start right away, you can connect your email account already.
**Warning:** Zammad reacts to fetched emails by default. If that’s not what you want, skip this step for now.

Learn more about the email channel within the documentation for email channels.

After finishing the wizard you’ll be automatically logged in to the just created account.

### 12.2 Further Steps

In our opinion the next step order would like below sample. You can skip parts you don’t need or adapt. All parts are described within Zammad’s admin documentation.

1. Configure your required groups
2. Adjust triggers as needed
3. Add postmaster filters if needed
4. Configure SLAs if needed
5. add email / social media channels & signatures
   (go back to group settings to add outgoing email addresses)
6. Add Text Modules
7. Add Organizations
8. Configure roles if needed
9. Consider Third Party logins or LDAP integration for easier logins
10. Add agent accounts (users)
11. Consider backup strategies for Zammad. See *Backup and Restore*

From point 5 on you’ll be able to work productive in theory.
**Hint:** Are you still lost?

If above list doesn’t help you or you’ll need to jump in a lot faster, you can also get Workshops with one of our Zammad consultants.
CHAPTER
THIRTEEN

MIGRATING TO ZAMMAD

Zammad will migrate the following information:

- Tickets and their Articles
- Groups / Queues
- Organizations
- Agents and Customers (if applicable)

After migrating to Zammad you’ll want to continue with the *First steps* to configure Zammad. This has to be done *after migration*.

13.1 Limitations

There might be source dependent limitations which we will be covering on the direct migration pages. However, these limitations count for all migrations:

- Migrations are only possible on new instances.
- Migrations are only possible from *one sources*. Several migration sources on one instance are *not* supported.
- Zammad can’t migrate object types it doesn’t know, migrations will fail.
- Zammad migrates *all or nothing*. This means that you can’t deselect specific information specific groups, tickets or users.

13.2 Available Migration Options

13.2.1 From Freshdesk

Limitations

Please note below Freshdesk specific limitations. These are additional limitations to the *general ones listed*.

- Differential migrations are *not* supported!
  The general suggestion is to run a test import before to learn how long the migration is going to take.
- **Important**: Please note that migration speed highly depends on your Freshdesk plan (API rate limits apply).
- Due to API limitations Zammad will not show the total number of objects to import, but instead correct them in steps of 100.
• User passwords are not migrated and will require the user to use the password reset link on the login page.

**Note:** Your Freshdesk plan has to provide API support. This may not apply to all available plans.

---

**Prerequisites**

Zammad requires API access which is why you’ll need to create an API key for the migration. The migrator will request your Freshdesk subdomain and API key.

**Warning:** To be or not to be

Ensure to retrieve the API key with a full administrator account. Less privileged users will end in a broken migration.

---

**Importing Freshdesk data**

Generally you have two options on how to migrate data. If you have a fairly big instance with a lot of data, you may want to consider using the console over the browser version.

Via browser

Via console

After installing Zammad and configuring your webserver, navigate to your Zammads FQDN in your browser and follow the migration wizard.

Depending on the number of users, tickets and Freshdesk plan this may take some while.

---

**Note:** Scheduler got interrupted

---

**Freshdesk Migration**

Interrupted by scheduler restart. Please restart manually or wait till next execution time.

If this message appears after providing your credentials, please be patient. The migration should start within 5 minutes.

If you receive above message after the migration begun, please consider using the console approach instead and reset the installation.

---

**Hint:** Here’s how to open a console.
# package installation
$ zammad run rails c

# source installation
$ rails c

Learn more about the Zammad console.

To prepare the migration, run the following commands

```ruby
# Set variables for easier settings
$ subdomain = '{freshdesk subdomain}.freshdesk.com'
$ token = '{freshdesk token}'

# Update Zammad settings for freshdesk import
$ Setting.set('import_freshdesk_endpoint', "https://#{subdomain}/api/v2")
$ Setting.set('import_freshdesk_endpoint_key', token)
$ Setting.set('import_backend', 'freshdesk')
$ Setting.set('import_mode', true)
```

**Hint:** Want to know if your configuration works before hand? Run the following command:

```ruby
Sequencer.process('Import::Freshdesk::ConnectionTest')
```

To start the actual migration, run the following commands

```ruby
# That the actual job
$ job = ImportJob.create(name: 'Import::Freshdesk')
$ AsyncImportJob.perform_later(job)
```

**Tip:** Want to check the state of the migration?

Running the following command in a rails console will provide detailed state information of your migration.

```ruby
pp ImportJob.find_by(name: 'Import::Freshdesk')
```

To give you an idea how the migration job state looks like, you can use below tabs. As long as finished_at is nil, the process is still running.

- Freshly started import
- Import half way
- Finished import

```ruby
<ImportJob:0x0000000008274310
 id: 1,
 name: "Import::Freshdesk",
 dry_run: false,
 payload: {},
 result: {
 "Organizations"=>

```
#<ImportJob:0x000055ba3d9dbbb8
id: 1,
name: "Import::Freshdesk",
dry_run: false,
payload: {},
result:
{
"Groups"=>
{
"skipped"=>0,
"created"=>3,
"updated"=>0,
"unchanged"=>0,
"failed"=>0,
"deactivated"=>0,
"sum"=>3,
"total"=>3},
"Organizations"=>
{
"skipped"=>0,
"created"=>193,
"updated"=>1,
"unchanged"=>0,
"failed"=>0,
"deactivated"=>0,
"sum"=>194,
"total"=>194},
"Users"=>
{
"skipped"=>0,
"created"=>3352,
"updated"=>0,
"unchanged"=>0,
"failed"=>0,
"deactivated"=>0,
"sum"=>3352,
"total"=>3352},
"Tickets"=>
{
"skipped"=>0,
"created"=>987,
"updated"=>0,
"unchanged"=>0,
"failed"=>0,
"deactivated"=>0,
"sum"=>987,
"total"=>987},
"Tickets"=>
{
"skipped"=>0,
"created"=>987,
"updated"=>0,
"unchanged"=>0,
"failed"=>0,
"deactivated"=>0,
"sum"=>987,
"total"=>987},
"deactivated"=>0,
"sum"=>987,
"total"=>1000},
started_at: Tue, 04 Jan 2022 11:37:38 UTC +00:00,
finished_at: nil,
created_at: Tue, 04 Jan 2022 11:37:36 UTC +00:00,
updated_at: Tue, 04 Jan 2022 12:12:52 UTC +00:00>

#<ImportJob:0x0000561da0def350
id: 1,
name: "Import::Freshdesk",
dry_run: false,
payload: {},
result: 
{"Groups"=>
{"skipped"=>0,
"created"=>3,
"updated"=>0,
"unchanged"=>0,
"failed"=>0,
"deactivated"=>0,
"sum"=>3,
"total"=>3},
"Organizations"=>
{"skipped"=>0,
"created"=>193,
"updated"=>1,
"unchanged"=>0,
"failed"=>0,
"deactivated"=>0,
"sum"=>194,
"total"=>194},
"Users"=>
{"skipped"=>0,
"created"=>3352,
"updated"=>0,
"unchanged"=>0,
"failed"=>0,
"deactivated"=>0,
"sum"=>3352,
"total"=>3352},
"Tickets"=>
{"skipped"=>0,
"created"=>4714,
"updated"=>0,
"unchanged"=>0,
"failed"=>1,
"deactivated"=>0,
"sum"=>4715,
"total"=>4715},
started_at: Tue, 04 Jan 2022 11:37:38 UTC +00:00,
finished_at: Tue, 04 Jan 2022 14:30:57 UTC +00:00,

13.2. Available Migration Options
After the import has finished, run the following commands

```bash
$ Setting.set('import_mode', false)
$ Setting.set('system_init_done', true)
$ Cache.clear
```

**After migration**

As the migration technically skips the getting started wizard, please note that you want to adjust your FQDN settings (FQDN & HTTP-Type).

**Hint:** How do I login?

Zammad provides admin access to the user whose API token you provided. Use the admins email address and api token provided during the migration to login.

All other users will have to use the password reset function or login methods like LDAP or one click logins.

After successfully migrating your Freshdesk instance, continue with First steps.

**Restarting from scratch**

Turned wrong at some point? You can find the required commands to reset Zammad in our Dangerzone.

**13.2.2 From Kayako**

**Limitations**

Please note below Kayako specific limitations. These are additional limitations to the general ones listed.

- Differential migrations are not supported!
  The general suggestion is to run a test import before to learn how long the migration is going to take.
- Selfhosted installations (Kayako classic) are not supported.
- The following ticket field customizations are being ignored (affects “Scale” plan):
  - Custom ticket states,
  - Custom ticket priorities, and
  - Custom ticket types.

- **Important:** Please note that migration speed highly depends on your Kayako plan (API rate limits apply).
- User passwords are not migrated and will require the user to use the password reset link on the login page.

**Note:** Your Kayako plan has to provide API support. This may not apply to all available plans.
**Prerequisites**

Zammad requires API access which is why the migrator will request your Kayako-URL, email address and password.

**Warning:** To be or not to be

Ensure to provide an user account with **full administrative** permissions. Less privileged users will end in a broken migration.

**Importing Kayako data**

Generally you have two options on how to migrate data. If you have a fairly big instance with a lot of data, you may want to consider using the console over the browser version.

Via browser

Via console

After installing Zammad and configuring your *webserver*, navigate to your Zammads FQDN in your browser and follow the migration wizard.

Depending on the number of users, tickets and Kayako plan this may take some while.

**Note:** Scheduler got interrupted

If this message appears after providing your credentials, please be patient. The migration should start within 5 minutes.

If you receive above message after the migration begun, please consider using the console approach instead and reset the installation.

**Hint:** Here’s how to open a console.

```bash
# package installation
$ zammad run rails c

# source installation
$ rails c
```

Learn more about the **Zammad console**.
To prepare the migration, run the following commands

```ruby
# Set variables for easier settings
$subdomain = '{kayako subdomain}.kayako.com'
$email = '{kayako admin email address}'
$password = '{kayako admin password}'

# Update Zammad settings for Kayako import
$Setting.set('import_kayako_endpoint', "https://#{subdomain}/api/v1")
$Setting.set('import_kayako_endpoint_username', email)
$Setting.set('import_kayako_endpoint_password', password)
$Setting.set('import_backend', 'kayako')
$Setting.set('import_mode', true)
```

**Hint:** Want to know if your configuration works before hand? Run the following command:

```ruby
Sequencer.process('Import::Kayako::ConnectionTest')
```

To start the actual migration, run the following commands

```ruby
# That the actual job
$job = ImportJob.create(name: 'Import::Kayako')
AsyncImportJob.perform_later(job)
```

**Tip:** Want to check the state of the migration?

Running the following command in a rails console will provide detailed state information of your migration.

```ruby
pp ImportJob.find_by(name: 'Import::Kayako')
```

To give you an idea how the migration job state looks like, you can use below tabs. As long as `finished_at` is `nil`, the process is still running.

Freshly started import

Import half way

Finished import

```ruby
<ImportJob:0x0000d008274310
  id: 1,
  name: "Import::Kayako",
  dry_run: false,
  payload: {},
  result:
    "Organizations"=>
      {"skipped"=>0,
       "created"=>0,
       "updated"=>0,
       "unchanged"=>0,
       "failed"=>0,
       "deactivated"=>0,
       "sum"=>0,
      }
```
"total"=>100},

started_at: Mon, 03 Jan 2022 18:41:51 UTC +00:00,
finished_at: nil,
created_at: Mon, 03 Jan 2022 18:41:16 UTC +00:00,
updated_at: Mon, 03 Jan 2022 18:43:32 UTC +00:00>

#<ImportJob:0x000055ba3d9dbbb8
id: 1,
name: "Import::Kayako",
dry_run: false,
payload: {},
result:
{"Groups"=>
 {"skipped"=>0,
  "created"=>3,
  "updated"=>0,
  "unchanged"=>0,
  "failed"=>0,
  "deactivated"=>0,
  "sum"=>3,
  "total"=>3},
"Organizations"=>
 {"skipped"=>0,
  "created"=>193,
  "updated"=>1,
  "unchanged"=>0,
  "failed"=>0,
  "deactivated"=>0,
  "sum"=>194,
  "total"=>194},
"Users"=>
 {"skipped"=>0,
  "created"=>3352,
  "updated"=>0,
  "unchanged"=>0,
  "failed"=>0,
  "deactivated"=>0,
  "sum"=>3352,
  "total"=>3352},
"Tickets"=>
 {"skipped"=>0,
  "created"=>987,
  "updated"=>0,
  "unchanged"=>0,
  "failed"=>0,
  "deactivated"=>0,
  "sum"=>987,
  "total"=>1000},

started_at: Tue, 04 Jan 2022 11:37:38 UTC +00:00,
finished_at: nil,
created_at: Tue, 04 Jan 2022 11:37:36 UTC +00:00,
updated_at: Tue, 04 Jan 2022 12:12:52 UTC +00:00>

13.2. Available Migration Options
After the import has finished, run the following commands

```bash
Setting.set('import_mode', false)
Setting.set('system_init_done', true)
Cache.clear
```
**After migration**

As the migration technically skips the getting started wizard, please note that you want to adjust your FQDN settings (FQDN & HTTP-Type).

**Hint: How do I login?**

Zammad provides admin access to the user whose login credentials you provided. Use the admins email address and password provided during the migration to login.

All other users will have to use the password reset function or login methods like LDAP or one click logins.

After successfully migrating your Kayako instance, continue with *First steps*.

**Restarting from scratch**

Turned wrong at some point? You can find the required commands to reset Zammad in our *Dangerzone*.

### 13.2.3 Migration from OTRS

**Limitations**

Please note below OTRS specific limitations. These are additional limitations to the *general ones listed*.

- Password migration works for OTRS >= 3.3 only  
  (on older instances a password reset within Zammad will be required)
- If you plan to import a differential migration after, do not change any data in Zammad!
- Only customers of tickets are imported
- Zammad expects your OTRS timestamps to be UTC and won’t adjust them
- If you plan to import a differential after, **do not** change any data in Zammad!

**Note:** Supported OTRS version: **3.1 up to 6.x**

**Prerequisites**

**Step 1: Install Znuny4OTRS-Repo**

This is a dependency of the OTRS migration plugin.

- OTRS 6
- OTRS 5
- OTRS 4
- OTRS 3

Step 2: Install OTRS migration plugin

OTRS 6
OTRS 5
OTRS 4
OTRS 3

Hint: In some cases restarting your webserver may help to solve internal server errors.

Importing OTRS data

via Browser
via Console

Note: If your OTRS installation is rather huge, you might want to consider using the command line version of this feature. This also applies if you experience Timeouts during the migration.

After installing Zammad and configuring your webserver, navigate to your Zammads FQDN in your Browser and follow the migration wizard.

Depending on the size of your OTRS installation this may take a while.

You can get an idea of this process in the migrator video on vimeo.

Hint: Here's how to open a console.

# package installation
$ zammad run rails c

# source installation
$ rails c
Learn more about the Zammad console.

If you miss this at the beginning or you want to re-import again you have to use the command line at the moment. Stop all Zammad processes and switch Zammad to import mode (no events are fired - e. g. notifications, sending emails, ...)

**Start the migration** Ensure to replace `xxx` with your values.

```
>> Setting.set('import_otrs_endpoint', 'https://xxx/otrs/public.pl?
  Action=ZammadMigrator')
>> Setting.set('import_otrs_endpoint_key', 'xxx')
>> Setting.set('import_mode', true)
>> Import::OTRS.start
```

After the import has finished, run the following commands

```
$ Setting.set('import_mode', false)
$ Setting.set('system_init_done', true)
$ Cache.clear
```

After successfully migrating your OTRS installation, continue with *First steps.*

**Importing a differential**

**Note:** This is only possible after finishing an earlier OTRS import **successful**.

In some cases it might be desirable to update the already imported data from OTRS. This is possible with the following commands.

**Run a differential import**

```
>> Setting.set('import_otrs_endpoint', 'http://xxx/otrs/public.pl?
  Action=ZammadMigrator')
>> Setting.set('import_otrs_endpoint_key', 'xxx')
>> Setting.set('import_mode', true)
>> Setting.set('system_init_done', false)
>> Import::OTRS.diff_worker
```

After the import has finished, run the following commands

```
$ Setting.set('import_mode', false)
$ Setting.set('system_init_done', true)
$ Cache.clear
```

All changes that occurred after your first migration should now also be available within your Zammad installation.
Zammad

Restarting from scratch

Turned wrong at some point? You can find the required commands to reset Zammad in our Dangerzone.

13.2.4 From Zendesk

Limitations

Please note below Zendesk specific limitations. These are additional limitations to the general ones listed.

• Differential migrations are not supported!
  The general suggestion is to run a test import before to learn how long the migration is going to take.

• Important: Please note that migration speed highly depends on your Zendesk plan (API rate limits apply).

• User passwords are not migrated and will require the user to use the password reset link on the login page.

  Note: Your Zendesk plan has to provide API support. This may not apply to all available plans.

Prerequisites

Zammad requires API access which is why you’ll need to create an API key for the migration. The migrator will request your Zendesk-URL, email address and API key.

  Warning: To be or not to be
  Ensure to retrieve the API key with a full administrator account. Less privileged users will end in a broken migration.

Importing Zendesk data

Generally you have two options on how to migrate data. If you have a fairly big instance with a lot of data, you may want to consider using the console over the browser version.

Via browser
Via console

After installing Zammad and configuring your webserver, navigate to your Zammad’s FQDN in your browser and follow the migration wizard.

Depending on the number of users, tickets and Zendesk plan this may take some while.

  Note: Scheduler got interrupted
If this message appears after providing your credentials, please be patient. The migration should start within 5 minutes.

If you receive above message after the migration begun, please consider using the console approach instead and reset the installation.

**Hint:** Here’s how to open a console.

```bash
# package installation
$ zammad run rails c

# source installation
$ rails c
```

Learn more about the Zammad console.

To prepare the migration, run the following commands

```ruby
# Set variables for easier settings
$ subdomain = '{zendesk url}'
$ email = '{zendesk admin email address}'
$ token = '{zendesk token}'

# Update Zammad settings for Zendesk import
$ Setting.set('import_zendesk_endpoint', "https://#{subdomain}/api/v2")
$ Setting.set('import_zendesk_endpoint_username', email)
$ Setting.set('import_zendesk_endpoint_key', token)
$ Setting.set('import_backend', 'zendesk')
$ Setting.set('import_mode', true)
```

**Hint:** Want to know if your configuration works before hand? Run the following command:

```ruby
Sequencer.process('Import::Zendesk::ConnectionTest')
```

To start the actual migration, run the following commands

```ruby
# That the actual job
$ job = ImportJob.create(name: 'Import::Zendesk')
$ AsyncImportJob.perform_later(job)
```
Tip: Want to check the state of the migration?

Running the following command in a rails console will provide detailed state information of your migration.

```
pp ImportJob.find_by(name: 'Import::Zendesk')
```

To give you an idea how the migration job state looks like, you can use below tabs. As long as `finished_at` is `nil`, the process is still running.

Freshly started import

Import half way

Finished import

```
#<ImportJob:0x00000000008274310
  id: 1,
  name: "Import::Zendesk",
  dry_run: false,
  payload: {},
  result: {
    "Organizations"=>
      {
        "skipped"=>0,
        "created"=>0,
        "updated"=>0,
        "unchanged"=>0,
        "failed"=>0,
        "deactivated"=>0,
        "sum"=>0,
        "total"=>100},
    started_at: Mon, 03 Jan 2022 18:41:51 UTC +00:00,
    finished_at: nil,
    created_at: Mon, 03 Jan 2022 18:41:16 UTC +00:00,
    updated_at: Mon, 03 Jan 2022 18:43:32 UTC +00:00>
```

```
#<ImportJob:0x000000000055ba3d9dbbb8
  id: 1,
  name: "Import::Zendesk",
  dry_run: false,
  payload: {},
  result: {
    "Groups"=>
      {
        "skipped"=>0,
        "created"=>3,
        "updated"=>0,
        "unchanged"=>0,
        "failed"=>0,
        "deactivated"=>0,
        "sum"=>3,
        "total"=>3},
    "Organizations"=>
      {
        "skipped"=>0,
        "created"=>193,
        "updated"=>1,
        "total"=>193},
    started_at: Mon, 03 Jan 2022 18:41:51 UTC +00:00,
    finished_at: nil,
    created_at: Mon, 03 Jan 2022 18:41:16 UTC +00:00,
    updated_at: Mon, 03 Jan 2022 18:43:32 UTC +00:00>
```

(continues on next page)
<table>
<thead>
<tr>
<th>Category</th>
<th>Skipped</th>
<th>Created</th>
<th>Updated</th>
<th>Unchanged</th>
<th>Failed</th>
<th>Deactivated</th>
<th>Sum</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Groups</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Organizations</td>
<td>0</td>
<td>193</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>194</td>
<td>194</td>
</tr>
</tbody>
</table>

---

13.2. Available Migration Options
After the import has finished, run the following commands

```
Setting.set('import_mode', false)
Setting.set('system_init_done', true)
Cache.clear
```

**After migration**

As the migration technically skips the getting started wizard, please note that you want to adjust your FQDN settings (FQDN & HTTP-Type).

**Hint:** How do I login?

Zammad provides admin access to the user whose API token you provided. Use the admins email address and api token provided during the migration to login.

All other users will have to use the password reset function or login methods like LDAP or one click logins.

After successfully migrating your Zendesk instance, continue with *First steps.*
**Restarting from scratch**

Turned wrong at some point? You can find the required commands to reset Zammad in our *Dangerzone*.

---

**Note:** Missing a migration source?

If we don’t cover your favorite source yet, you’ll have two options. You can either fiddle around by using Zammad’s powerful *API* or drop our *sales team* a message for a custom development or even migrator sponsoring.

*Migrations are available for hosted setups too, contact support for further information!*
Zammad uses Ruby on Rails so you can make use of the rails console.

**Warning:** Please double check your commands before running, as some of those commands might cause data loss or damaged tickets! If you’re unsure, use a test system first!

To open the rails console on the shell you have to enter the following commands.

### 14.1 Start Zammad’s Rails console

#### 14.1.1 Running a single command

The following command will allow you to run a single command, without running a shell (e.g. for automation).

**Note:** Replace `{COMMAND}` with your command you want to run.

**Tip:** If you enter a `p` in front of your command (e.g. like `rails r 'p Delayed::Job.count'`), you’ll actually receive a printed output (without you won’t!).

```
# package installation
$ zammad run rails r '{COMMAND}''

# source installation
$ rails r '{COMMAND}''
```

#### 14.1.2 Running several commands in a shell

The following command will provide you a rails console. It allows you to run several commands inside it. This reduces loading times greatly.

```
# package installation
$ zammad run rails c
```
# source installation
$ rails c

14.2 Working on the console

Here’s a topic list for quick jumping and better overview.

14.2.1 Query and set / update Zammad settings

Note: Please note that this is not a full command list, if you’re missing commands, feel free to ask over at the Community.

ticket_hook setting

This will give you the ticket hook that you’ll find inside the [] in front of the ticket number. By default this will be Ticket# - you shouldn’t change this setting in a productive system.

```ruby
>> Setting.get('ticket_hook')
```

FQDN setting

Get the current FQDN setting of Zammad and, if needed, adjust it.

Note: This setting has no effect on SSL certificates or any web server configurations.

```ruby
>> Setting.get('fqdn') # Get current FQDN
>> Setting.set('fqdn', 'new.domain.tld') # Set a new FQDN
```

HTTP(s) setting

This setting indirectly belongs to your FQDN setting and is relevant for variable based URLs (e.g. in notifications) Zammad generated.

Warning: This setting also affects Zammad’s CSRF token behavior. If you set this setting to e.g. HTTPs but you’re using HTTP, logging in will fail!

Note: This setting has no effect on SSL certificates or any web server configurations.

```ruby
>> Setting.get('http_type') # Get the current http type
>> Setting.set('http_type', 'https') # Change the http type to HTTPs
```
Storage provider setting

The storage provider setting is set to DB on default installations. However, if you receive a lot of attachments or have a fairly busy installation, using the database to store attachments is not the best approach.

Use the following command

```ruby
>> Setting.get('storage_provider')  # get the current Attachment-Storage
>> Setting.set('storage_provider', 'DB')  # Change Attachment-Storage to database
```

**Tip:** The following settings are available in a default installation:

- DB (database)
- File (Filesystem (/opt/zammad/storage/))

Configuring Elasticsearch

If your Elasticsearch installation changes, you can use the following commands to ensure that Zammad still can access Elasticsearch.

```ruby
>> Setting.set('es_url', 'http://127.0.0.1:9200')  # Change elasticsearch URL
>> Setting.set('es_user', 'elasticsearch')  # Change elasticsearch user
>> Setting.set('es_password', 'zammad')  # Change the elasticsearch password for authentication
>> Setting.set('es_index', Socket.gethostname + '_zammad')  # Change the index name
>> Setting.set('es_attachment_ignore', %w[.png .jpg .jpeg .mpeg .mpg .mov .bin .exe .box .mbox])  # A list of ignored file extensions (they will not be indexed)
>> Setting.set('es_attachment_max_size_in_mb', 50)  # Limit the Attachment-Size to push to your elasticsearch index
```

Enable proxy

Zammad needs to use a proxy for network communication? Set it here.

```ruby
>> Setting.set('proxy', 'proxy.example.com:3128')
>> Setting.set('proxy_username', 'some user')
>> Setting.set('proxy_password', 'some pass')
```

14.2.2 Advanced customization settings

On this page you can find some settings that you won’t find within the Zammad UI. Those settings might come in handy as it can change Zammad’s behavior.

**Note:** Please note that this is not a full command list, if you’re missing commands, feel free to ask over at the Community.
Send all outgoing E-Mails to a BCC-Mailbox

This option allows you to send all outgoing E-Mails (not notifications) to a specific mailbox. Please note that this shouldn’t be a mailbox you’re importing already! This will apply to all groups and is a global setting.

```ruby
>> Setting.set('system_bcc', 'alias@domain.tld')
```

You can easily check the current BCC-Setting by running the following:

```ruby
>> Setting.get('system_bcc')
```

Activate counter on grouped overviews

This is a hidden setting which you can only set via Command-Line. This will globally enable a ticket number value in each heading for grouped elements.

```ruby
>> Setting.set('ui_table_group_by_show_count', true) # enable counter on grouped overviews
>> Setting.set('ui_table_group_by_show_count', false) # disable counter on grouped overviews
>> Setting.get('ui_table_group_by_show_count') # get current setting ('nil' is false)
```

Default ticket type on creation

Zammad allows you to define the default article type upon ticket creation. By default this will be a incoming phone call.

You can choose between

- phone-in (incoming call, default),
- phone-out (outgoing call) and
- email-out (Sending an E-Mail out).

```ruby
>> Setting.set('ui_ticket_create_default_type', 'email-out')
```

To check what setting is set currently, simply run:

```ruby
>> Setting.get('ui_ticket_create_default_type')
```
Adding a warning to the ticket creation process

If in case you need to give your agent a note or warning during ticket creation, you can do so with the below command. You can use three different warnings for

- Incoming Calls : 
  ```ruby
  phone-in"=>"
  ```
- Outgoing Calls : 
  ```ruby
  phone-out"=>"
  ```
- Outgoing E-Mails : 
  ```ruby
  email-out"=>"
  ```

```ruby
>> Setting.set('ui_ticket_create_notes', {
  'phone-in' => "You’re about to note a incoming phone call.",
  'phone-out' => "You’re about to note an outgoing phone call.",
  'email-out' => "You’re going to send out an E-Mail."
})
```

**Note:** You can use those three sub-settings independently, if you e.g. don't need a warning on incoming calls, simply leave out : 

```ruby
"phone-in"=>"
``` 
out of the setting. The setting itself is done within an array (`{}`).

To check what’s currently set, you can use:

```ruby
>> Setting.get('ui_ticket_create_notes')
```

Sample of the above setting:

### Show Email address of customer on customer selection (ticket creation)

By default Zammad will not display the E-Mail-Addresses of customers. The below option allows you to change this behavior.

```ruby
>> Setting.set('ui_user_organization_selector_with_email', true)
```

Get the current state of this setting with:

```ruby
>> Setting.get('ui_user_organization_selector_with_email')
```

### Change font settings for outgoing HTML mails

**Note:** Some clients (like Outlook) might fallback to other settings while it might work for other clients.

The below setting allows you to adjust Zammad’s email font setting. This setting does not require a service restart.

```ruby
>> Setting.set("html_email_css_font", "font-family:'Helvetica Neue', Helvetica, Arial,\n..., Geneva, sans-serif; font-size: 12px;")
```

If you want to check the current setting, you can simply run the below code.

```ruby
>> Setting.get('html_email_css_font')
```
14.2.3 Working on user information

Note: Please note that this is not a full command list, if you’re missing commands, feel free to ask over at the Community.

Find user

In order to work on user information or to check for specific information, you’ll need to find it first.

```ruby
>> User.find(4)          # We already know the ID of the user
>> User.find_by(email: 'your@email')       # Searching for the user by his Email address
>> User.find_by(login: 'john.doe')        # Searching for the user by his login
```

Unlock a locked user account

Tip: Unlocking a locked user account is also supported by Zammad’s web UI. Please refer the admin documentation for more information.

It sometimes happens that a user locks himself out by wildly trying the wrong password multiple times. Depending on your maximum failing login count (default: 10 times), Zammad might lock the account.

The user can’t login any more (forever) if he doesn’t change the password or you reset the counter.

```ruby
>> u=User.find(**USERID**)  
>> u.login_failed=0
>> u.save!
```

You can also double check if the account is locked by running the following (result needs to be 1 above your limit, so 11 for the default of 10 failing logins)

```ruby
>> User.find(**USERID**).login_failed
```

Change / Update Email address of user

If needed, you can simply change the Email address of the user.

Note: Please note that the login attribute is not affected by this and Zammad thus might show different information within the UI.

```ruby
>> u = User.find(**USERID**)  
>> u.email = 'user@example.com'
>> u.save!
```

You need to find the user ID of the user first for this.
Change / Update Login name of user

Change the user name of the user (e.g. if you want to login with a shorter username instead of a mail address)

```ruby
>> u = User.find(**USERID**)
>> u.login = 'user@example.com'
>> u.save!
```

You need to find the user ID of the user first for this.

Set admin rights for user

Don’t have access to Zammad anymore? Grant yourself or another user administrative rights.

```ruby
>> u = User.find_by(email: 'you@example.com')
>> u.roles = Role.where(name: ['Agent', 'Admin'])
>> u.save!
```

Set password for user

You or the user did forget his password? No problem! Simply reset it by hand if needed.

```ruby
>> User.find_by(email: 'you@example.com').update!(password: 'your_new_password')
```

Remove password for user

If you added a second authentication method (e.g. LDAP) after launch, there still may be a password in Zammad’s own user management. In cases like that users will be able to login with their (local) Zammad password in addition to the credentials stored on the external authentication provider. Simply remove the password stored by Zammad.

```ruby
>> User.find_by(email: 'you@example.com').update!(password: nil)
```

14.2.4 Working with ticket information

Note: Please note that this is not a full command list, if you’re missing commands, feel free to ask over at the Community.

Get the RAW mail that Zammad fetched

The following command will help you to check on received EML-files Zammad fetched. This comes in handy if you delete Mails upon fetching and you need to check the EML-file itself.

To get the first articles EML-file, you can use the following command. In our example the ticket number in question is 101234.

```ruby
>> Ticket.find_by(number: '101234').articles.first.as_raw.content
```
If needed, you can also get the raw content of later articles (you’ll need to find the correct article though). Again, we expect 101234 to be our ticket number. In the first step we get all article IDs of the ticket, from the list we get, we can then get the articles content.

```ruby
>> Ticket.find_by(number: '101234').article_ids
=> [4, 3, 2]
>> Ticket::Article.find(3).as_raw.content
```

**Note:** If you just use `Ticket::Article.find(3)` you can see further information (like who sent the mail, when we fetched it,...).

---

**Update all tickets of a specific customer**

**Warning:** Please note that this action can be expensive resource wise, if you have many tickets, this might slow down Zammad.

```ruby
>> Ticket.where(customer_id: 4).update_all(customer_id: 1)
```

---

**Priorities**

Ticket priorities help your agent to see how important a customer request is. Priorities are not available to customers and, Core wise, have no impact on how Zammad handles a ticket. You can however adjust Zammad’s behavior with e.g. triggers, SLAs and schedulers.

Not sure what priorities are available in the system? Either have a look in any ticket or run the following command.

```ruby
>> Ticket::Priority.pluck(:name)
```

---

**Adding priorities for tickets**

Ticket priorities come with several attributes, however, the most relevant as of now are: name, default_create and ui_color.

**Warning:** default_create allows you to define the default priority Zammad should use during ticket creation. **However** - on default installations this is the priority 2 normal. **You cannot have more than one priority as the default_create priority!**

**Note:** ui_color defines the CSS class to use. On default installations you can either use low-priority (light blueish) or important (redish). This affects how Zammad displays the ticket titles in overviews.

```ruby
>> Ticket::Priority.create(
  name: '4 super high',
  default_create: false,
) (continues on next page)
```
Change priority

If needed you can also set priorities to inactive or rename them if they don’t fit your desired scheme. Renaming would look like so:

```ruby
>> Ticket::Priority.update(name: '1 high')
```

Get ticket state types

This will show all state types needed for creating new ticket states.

**Tip:** What are state types?

Zammad uses state types to know what it should do with your state. This allows you to have different types like pending actions, pending reminders or closed states.

State types also indicate the color scheme to be used. You can learn more about that in our user documentation.

```ruby
>> Ticket::StateType.pluck(:id, :name)
```

Above will return both, the type ID and name - e.g.: `[[1, "new"], [2, "open"], ....

Add new ticket state

**Note:** Missing States you just created?

You might want to use `Ticket::State.pluck(:id, :name)` to get a listing of all available ticket states.

**Tip:** ignoring escalations

You can use `ignore_escalation: true`, to ignore possible SLA calculations (pending reminder and pending close do this by default).
Non-Pending states

A state that’s not a pending state (e.g. open, closed). Just replace 'open' by whatever you need (like closed).

```ruby
>> Ticket::State.create_or_update(
   name: 'Developing',
   state_type: Ticket::StateType.find_by(name: 'open'),
   created_by_id: 1,
   updated_by_id: 1,
)
```

Pending reminders

A pending reminder state that will send a reminder notification to the agent if the time has been reached.

```ruby
>> Ticket::State.create_or_update(
   name: 'pending customer feedback',
   state_type: Ticket::StateType.find_by(name: 'pending reminder'),
   ignore_escalation: true,
   created_by_id: 1,
   updated_by_id: 1,
)
```

Pending Action

A pending action that will change to another state if “pending till” has been reached.

```ruby
>> Ticket::State.create_or_update(
   name: 'pending and reopen',
   state_type: Ticket::StateType.find_by(name: 'pending action'),
   ignore_escalation: true,
   next_state: Ticket::State.find_by(name: 'open'),
   created_by_id: 1,
   updated_by_id: 1,
)
```

(Optional) Disable date and time picker (pending till) for pending states

Starting with Zammad 5.0, Core Workflows automatically handles displaying the “pending till” field for pending states. Below snippet is not required and is only relevant if you don’t want to create a workflow within the UI of Zammad.

Replace pending customer feedback with the pending state of your choice.

```ruby
>> CoreWorkflow.create_if_not_exists(
   name: 'remove pending till on state "pending customer feedback"',
   object: 'Ticket',
   condition_selected: { 'ticket.state_id'=>{ 'operator' => 'is', 'value' => Ticket::State.find_by(name: 'pending customer feedback').id.to_s } },
   perform: { 'ticket.pending_time'=> { 'operator' => 'remove', 'remove' => true } },
)```
Make new states available to UI

Before being able to use the new states within the WebApp, you need to run the following commands to make them available.

**Warning:** Please **do not replace** anything below, state_id is a named attribute which is correct and shall not be replaced!

```ruby
>> attribute = ObjectManager::Attribute.get(
  object: 'Ticket',
  name: 'state_id',
)
>> attribute.data_option[:filter] = Ticket::State.by_category(:viewable).pluck(:id)
>> attribute.screens[:create_middle]['ticket.agent'][:filter] = Ticket::State.by_category(:viewable_agent_new).pluck(:id)
>> attribute.screens[:create_middle]['ticket.customer'][:filter] = Ticket::State.by_category(:viewable_customer_new).pluck(:id)
>> attribute.screens[:edit]['ticket.agent'][:filter] = Ticket::State.by_category(:viewable_agent_edit).pluck(:id)
>> attribute.screens[:edit]['ticket.customer'][:filter] = Ticket::State.by_category(:viewable_customer_edit).pluck(:id)
>> attribute.save!
```

Limit available states for customers

**Tip:** Core Workflows allows you to achieve below described behavior any time without any issues. No need to use the console if you don’t want to!

By default Zammad allows customers to change Ticket states to open and closed. If this does not meet your requirements, you can adjust this at anytime. The below example shows how to restrict your customer to only close tickets if needed:

```ruby
>> attribute = ObjectManager::Attribute.get(
  object: 'Ticket',
  name: 'state_id',
)
>> attribute.screens[:edit]['ticket.customer'][:filter] = Ticket::State.where(name: ['closed']).pluck(:id)
>> attribute.save!
```

**Hint:** If you want to allow several different states for customers, you need to provide the state names as array - like

14.2. Working on the console
so: ['closed', 'open', 'my-amazing-state'] (instead of ['closed']).

You can check the current active states that customers can set like so:

```perl
>> ObjectManager::Attribute.get(
    object: 'Ticket',
    name: 'state_id',
  ).screens['edit']['ticket.customer']['filter']
```

The above will return one or more IDs - if you’re not sure which state they belong to, you can check the state name with the following command. (Ensure to replace `{ID}` with your returned ID(s))

```perl
>> Ticket::State.find({ID}).name
```

### 14.2.5 Working with ticket articles

**Note:** Please note that this is not a full command list, if you’re missing commands, feel free to ask over at the Community.

#### Count Public “Notes” toward SLAs

Normally, notes don’t count toward service-level agreements. Use the following command to include publicly-visible notes when tracking SLA compliance. (Internal notes will never affect SLA calculations.)

**Note:** By default, customers are not notified when public notes are added to a ticket. Set up a trigger if you wish to change this behavior.

**Warning:** Changing this setting will disable the option to delete public notes.

```perl
>> Ticket::Article::Type.find_by(name: 'note').update!(communication: true)  # Enable SLA to count notes as communication
>> Ticket::Article::Type.find_by(name: 'note').update!(communication: false)  # Enable SLA to ignore notes as communication
```

### 14.2.6 Working with groups

**Note:** Please note that this is not a full command list, if you’re missing commands, feel free to ask over at the Community.

To open the rails console on the shell you have to enter the following commands.
Find group

```
>> Group.find_by(name: 'Users').follow_up_possible
```

### 14.2.7 Working with chat logs

**Note:** Please note that this is not a full command list, if you’re missing commands, feel free to ask over at the Community.

#### Removing IP address logs

Use the following command to remove all IP address records from closed chats that haven’t been updated in the last seven days:

```
>> Chat::Session.where(state: 'closed').where('updated_at < ?', 7.days.ago).each do
    next if session.preferences['remote_ip'].blank?
    session.preferences.delete('geo_ip')
    session.preferences.delete('remote_ip')
    session.save!(touch: false)
end
```

#### 14.2.8 Other useful commands

**Note:** Please note that this is not a full command list, if you’re missing commands, feel free to ask over at the Community.

##### Fetch mails

The below command will do a manual fetch of mail channels. This will also show errors that might appear within that process.

```
>> Channel.fetch
```

##### Reprocess unprocessable mails

When Zammad encounters a mail it cannot parse (e.g. due to a parser bug or a malformed message), it will store the mail in `tmp/unprocessable_mail/<ID>.eml`, give up on attempting to parse the mail, and will warn on the monitoring page that there are unprocessed mails.

To force Zammad to reattempt to parse those mails, run the following command:

```
>> Channel::EmailParser.process_unprocessable_mails
```
Zammad

In case of a malformed message (e.g. an invalid email address in one of the header fields), you may need to manually edit the mail before Zammad can process it.

If Zammad fails to process the message, it will remain in the tmp/unprocessable_mail folder; otherwise it will be removed after it has been parsed successfully.

Add translation

This comes in handy if you e.g. added a new state that you need to translate for several languages.

```ruby
>> Translation.create_if_not_exists(:locale => 'de-de', :source => "New", :target => "Neu →", created_by_id: 1, updated_by_id: 1)
```

**Warning:** While Zammad knows further attributes for the Translation model, please *do not* set them manually. Doing so may interfere with our Weblate translation process and cause you loosing your custom translations.

If you want to translate code base strings that are available within standard code, please use Weblate instead.

Translating attributes

By default Zammad will not translate custom attributes. With the following code you can enable translation. This will translate the attribute display name and the display names of values (if it’s a value field). For this to work, just replace `{attribute-name}` with the name of your attribute.

```ruby
>> attribute = ObjectManager::Attribute.find_by(name: '{attribute-name}')
>> attribute.data_option[:translate] = true  # set this to false to disable
              # translation again
>> attribute.save!
```

**Note:** Translating value display names works for the following attribute types:

- Boolean
- Select
- Tree Select

If you’re translating the display name of e.g. an Integer-attribute, this works as well!

Fill a test system with test data

**Danger:** Don’t run this in a productive environment! This can slow down Zammad and is hard to revert if you create much!

The below command will add 50 agents, 1000 customers, 20 groups, 40 organizations, 5 new overviews and 100 tickets. You can always use 0 to not create specific items. Zammad will create random data which make no logical sense.
14.2.9 Deleting records

**Danger:** The commands listed here cause irrecoverable data loss! Only proceed if you know what you’re doing and you have a backup!

**Note:** Please note that this is not a full command list, if you’re missing commands, feel free to ask over at the Community.

---

**Removing tickets (and their articles)**

```ruby
# Delete a ticket (specified by database ID)
>> Ticket.find(4).destroy

# Delete all tickets
>> Ticket.destroy_all

# Keep some tickets (specified by database ID); delete the rest
>> tickets_to_keep = [1, 2, 3]
>> Ticket.where.not(id: tickets_to_keep).destroy_all
```

---

**Removing users**

**Warning:** Customers may not be deleted while they have tickets remaining in the system.

As such, the examples below will delete not only the specified customers, but all tickets associated with them, as well. Below commands remove upon executing without any further warnings.

**Hint:** If you’re not sure what to do and need to learn more about what Zammad does upon removing users, please consider using Zammad’s UI options in stead.

Our documentation for the data privacy function will help you a lot!

Removing users is possible in 2 ways: A single user and in bulk.

Remove a single user

Remove several users

```ruby
>> User.find_by(email: '<email address>').destroy
```
Removing organizations

Note: Removing an organization does **not** delete associated customers.

**Step 1: Select organizations**

```ruby
# by "active" status
>> organizations = Organization.where( active: false)

# by name
>> organizations = Organization.where( name: 'Acme')

# by partial match on notes
>> organizations = Organization.where( 'note LIKE ?', '%foo%')
```

**Step 2: Preview affected organizations**

```ruby
>> puts organizations.map { |org| "ORGANIZATION #{org.name}" }.join("\n")
```

**Step 3: Proceed with deletion**

```ruby
>> organizations.each do |org|
  puts %{Preparing deletion of organization "#{org.name}"...}
  org.members.each do |member|
    puts " Removing #{member.fullname} from organization..."
    member.update!(organization_id: nil)
  end
  puts " Deleting #{org.name}"...
  org.destroy
end
```

Removing system records

```ruby
# Remove all online notifications
>> OnlineNotification.destroy_all

# Remove all entries from the Activity Stream (dashboard)
>> ActivityStream.destroy_all

# Remove entries for all recently viewed objects
# (tickets, users, organizations)
>> RecentView.destroy_all
```

(continues on next page)
Reset Zammad installation

**Hint:** Below commands are incomplete intentionally, error outputs will hint you through! The following operations will cause data loss and are for development / testing only.

Don’t forget to stop Zammad before trying to drop the database!

```bash
$ rake db:drop
$ rake db:create
$ rake db:migrate
$ rake db:seed
```
We would be glad if you contribute to Zammad. You can do this in several ways. Contributions are mainly done by forking one of our repos on GitHub and creating a pull request with your changes.

All repos can be found at https://github.com/zammad

### 15.1 Source Code

The Zammad source code can be found on GitHub at https://github.com/zammad/zammad

For more information on how to contribute to Zammad, please have a look at https://zammad.org/participate and at the Developer Manual.

### 15.2 Documentation

Do you want to contribute to the Zammad documentation?

Open a new GitHub pull request at

- https://github.com/zammad/zammad-documentation (this documentation)
- https://github.com/zammad/zammad-admin-documentation
- https://github.com/zammad/zammad-user-documentation

with your changes.

The Zammad documentation is hosted on Read the Docs. You can read it there at

- https://docs.zammad.org (this documentation)
- https://admin-docs.zammad.org
- https://user-docs.zammad.org

or browse the files via GitHub which also renders the used ReStructuredText markup.
15.2.1 ReStructuredText markup

If you like to edit the docs use the ReStructuredText markup language. Information about this language can be found at:

- http://docutils.sourceforge.net/docs/user/rst/quickref.html

Thanks!

Zammad Team
The main Zammad repository at https://github.com/zammad/zammad has several branches.

16.1 develop

- This is the current (unreleased) development state of next major release (this will become the new stable branch).
- Don’t use it for production!
- Supported with bug and security fixes - see also our Security Policy.

16.2 stable

- This is the current stable release, e.g. Zammad 5.2.
- Use this branch for production installations.
- Supported with bug and security fixes - see also our Security Policy.

16.3 stable-x.y

- These are the branches of old versions of Zammad like stable-5.1 for Zammad 5.1.
- No support for bug or security issues is provided.
Zammad provides a powerful REST-API which allows all operations that are available via UI as well. This page gives you a first impression for things that generally count for all endpoints and how to adapt.

### 17.1 API clients

There are API clients available. Please note that these clients may not provide access to all available endpoints listed here.

- Ruby Client *(Official)*
- PHP Client *(Official)*
- Python Client *(Third-Party)*
- .NET Client *(Third-Party)*
- Android API-Client *(Third-Party)*

**Note:** Please note that this is a API client only, it’s no “ready to use” App.

### 17.2 Authentication

Zammad supports three different authentication methods for its API.

**HTTP Basic Authentication (username/password)**

The username / password must be provided as HTTP header in the HTTP call.

This authentication method can be disabled and may not be available in your system.

```
$ curl -u {username}:{password} https://{fqdn}/{endpoint}
```

**Note:** We strongly suggest against using basic authentication. Use access tokens when ever possible!

**HTTP Token Authentication (access token)**

The access token must be provided as HTTP header in the HTTP call.

Each user can create several access tokens in their user preferences.

This authentication method can be disabled and may not be available in your system.
### OAuth2 (token access)

The token must be provided as HTTP header in your calls. This allows 3rd party applications to authenticate against Zammad.

```bash
$ curl -H "Authorization: Token token={your_token}" https://{fqdn}/{endpoint}
```

### 17.3 Endpoints and example data

For simplicity we’ll not provide specific commands on the next pages, but instead tell the possible call method (e.g. GET) and the endpoint to use (e.g. /api/v1/users). In case Zammad expects information within these endpoint urls, we’ll put them into curly braces like so: /api/v1/users/{user id}

The response format will be a complete JSON response from a default Zammad instance. Please keep in mind that you may see more fields or general information in case you added objects or other information.

### 17.4 Content Type

Zammad returns JSON payloads whenever you retrieve data. If you’re going to provide data, no matter of the general request type, don’t forget to provide the content type `application/json` as well.

### 17.5 Response Payloads (expand)

Zammad always returns information including hints to all relations. If you need more information than that (because IDs may not be enough) you can also extend your endpoint calls with `?expand=true`.

This switch will provide even more information — at least named relations on top of the ID ones. Below you can find two examples to compare - one for ticket and user each.

**User payload**

```json
{
    "active": true,
    "login_failed": 0,
    "verified": false,
    "source": null,
    "login": "chris@chrispresso.com",
    "last_login": "2021-09-23T13:17:24.817Z",
    "id": 3,
    "updated_by_id": 1,
    "organization_id": 2,
    "firstname": "Christopher",
    "lastname": "Miller",
}
```

**Ticket payload**

```json
{
    "active": true,
    "creation_time": "2021-09-23T13:17:24.817Z",
    "subject": "Ticket subject",
    "status": "open",
    "updated_time": "2021-09-23T13:17:24.817Z",
    "created_by_id": 1,
    "organization_id": 2,
    "tags": ["tag1", "tag2"],
    "ticket_type": "issue",
    "priority": 1,
    "due_date": null,
    "notes": null,
    "resolved_at": null,
    "closed_by_id": null,
    "closed_at": null,
    "assignee_id": null,
    "dependencies": null,
    "relations": null,
}
```
"email": "chris@chrispresso.com",
"image": "7a6a0d1d94ad2037153cf3a6c1b49a53",
"image_source": null,
"web": "",
"phone": "",
"fax": "",
"mobile": "",
"department": "",
"street": "",
"zip": "",
"city": "",
"country": "",
"address": "",
"vip": false,
"note": "",
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_office_replacement_id": null,
"preferences":
{
  "notification_config":
  {
    "matrix":
    {
      "create":
      {
        "criteria":
        {
          "owned_by_me": true,
          "owned_by_nobody": true,
          "subscribed": true,
          "no": true
        },
        "channel":
        {
          "email": true,
          "online": true
        }
      },
      "update":
      {
        "criteria":
        {
          "owned_by_me": true,
          "owned_by_nobody": true,
          "subscribed": true,
          "no": true
        },
        "channel":
        {
          "email": true,
          "online": true
        }
      }
    }
  }
}
"online": true
}
},
"reminder_reached":
{
  "criteria":
  {
    "owned_by_me": true,
    "owned_by_nobody": false,
    "no": true
  },
  "channel":
  {
    "email": true,
    "online": true
  }
},
"escalation":
{
  "criteria":
  {
    "owned_by_me": true,
    "owned_by_nobody": false,
    "no": true
  },
  "channel":
  {
    "email": true,
    "online": true
  }
}
},
"group_ids":
[
  "2",
  "1",
  "3"
]
},
"locale": "de-de",
".intro": true,
"notification_sound":
{
  "file": "Xylo.mp3",
  "enabled": true
},
"cti": true,
"tickets_closed": 0,
"tickets_open": 1
},
"created_by_id": 1,
"created_at": "2021-07-26T14:44:41.066Z",
(continues on next page)
"updated_at": "2021-09-23T13:17:24.825Z",
"role_ids":
[
    1,
    2
],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [
    1
],
"group_ids": {
    "1": [
        "full"
    ],
    "2": [
        "full"
    ],
    "3": [
        "full"
    ]
}
{
    "active": true,
    "login_failed": 0,
    "verified": false,
    "source": null,
    "login": "chris@chrispresso.com",
    "last_login": "2021-09-23T13:17:24.817Z",
    "id": 3,
    "updated_by_id": 1,
    "organization_id": 2,
    "firstname": "Christopher",
    "lastname": "Miller",
    "email": "chris@chrispresso.com",
    "image": "7a6a0d1d94ad2037153cf3a6c1b49a53",
    "image_source": null,
    "web": "",
    "phone": "",
    "fax": "",
    "mobile": "",
    "department": "",
    "street": ""
}
"zip": "",
"city": "",
"country": "",
"address": "",
"vip": false,
"note": "",
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_office_replacement_id": null,
"preferences": {
  "notification_config": {
    "matrix": {
      "create": {
        "criteria": {
          "owned_by_me": true,
          "owned_by_nobody": true,
          "subscribed": true,
          "no": true
        },
        "channel": {
          "email": true,
          "online": true
        }
      },
      "update": {
        "criteria": {
          "owned_by_me": true,
          "owned_by_nobody": true,
          "subscribed": true,
          "no": true
        },
        "channel": {
          "email": true,
          "online": true
        }
      }
    },
    "reminder_reached": {
      "criteria": {
        "owned_by_me": true,
        "owned_by_nobody": false,
        "subscribed": true,
        "no": true
      }
    }
  }
}
"no": true,
"channel":
{
  "email": true,
  "online": true
},
"escalation":
{
  "criteria":
  {
    "owned_by_me": true,
    "owned_by_nobody": false,
    "no": true
  },
  "channel":
  {
    "email": true,
    "online": true
  }
}
},
"group_ids":
[ 2,
  1,
  3
],
"locale": "de-de",
"intro": true,
"notification_sound":
{
  "file": "Xylo.mp3",
  "enabled": true
},
"cti": true,
"tickets_closed": 0,
"tickets_open": 1
},
"created_by_id": 1,
"created_at": "2021-07-26T14:44:41.066Z",
"updated_at": "2021-09-23T13:17:24.825Z",
"role_ids":
[ 1,
  2
],
"organization_ids":
[],
"authorization_ids":

{
  "id": 3,
  "group_id": 1,
  "priority_id": 2,
  "state_id": 4,
  "organization_id": 3,
  "number": "71003",
  "title": "Order 787556",
  "owner_id": 3,
  "customer_id": 7,
  "note": null,
  "first_response_at": null,
  "first_response_escalation_at": null,
  "first_response_in_min": null,
  "first_response_diff_in_min": null,
  "close_at": null,
  "close_escalation_at": null,
  "close_in_min": null,
  "close_diff_in_min": null,
  "update_escalation_at": null,
  "update_in_min": null,
  "update_diff_in_min": null,
  "last_contact_at": "2021-02-26T12:44:43.888Z",
  "last_contact_agent_at": "2021-02-26T12:44:43.888Z",
  "last_contact_customer_at": "2021-02-24T14:44:43.828Z",
  "last_owner_update_at": null,
  "create_article_type_id": 1,
  "create_article_sender_id": 2,
  "article_count": 2,
  "escalation_at": null,
  "pending_time": null,
  "type": null,
  "preferences": {},
  "updated_by_id": 4,
  "created_by_id": 7,
  "created_at": "2021-02-24T14:44:43.828Z",
  "updated_at": "2021-07-26T14:44:43.906Z"
}

(continues on next page)
"first_response_at": null,
"first_response_escalation_at": null,
"first_response_in_min": null,
"first_response_diff_in_min": null,
"close_at": null,
"close_escalation_at": null,
"close_in_min": null,
"close_diff_in_min": null,
"update_escalation_at": null,
"update_in_min": null,
"update_diff_in_min": null,
"last_contact_at": "2021-02-26T12:44:43.888Z",
"last_contact_agent_at": "2021-02-26T12:44:43.888Z",
"last_contact_customer_at": "2021-02-24T14:44:43.828Z",
"last_owner_update_at": null,
"create_article_type_id": 1,
"create_article_sender_id": 2,
"article_count": 2,
"escalation_at": null,
"pending_time": null,
"type": null,
"time_unit": null,
"preferences": {},
"updated_by_id": 4,
"created_by_id": 7,
"created_at": "2021-02-24T14:44:43.828Z",
"updated_at": "2021-07-26T14:44:43.906Z",
"article_ids": [
  5,
  6
],
"ticket_time_accounting_ids": [],
"group": "Sales",
"organization": "Awesome Customer Inc.",
"ticket_time_accounting": [],
"state": "closed",
"priority": "2 normal",
"owner": "chris@chrispresso.com",
"customer": "samuel@example.com",
"created_by": "samuel@example.com",
"updated_by": "jacob@chrispresso.com",
"create_article_type": "email",
"create_article_sender": "Customer"

Warning: Please note that Core Workflows may restrict access to attributes or values. see Core Workflows limitations to learn more.
17.6 Pagination

As Zammad limits the number of returned objects for performance reasons, you may have to use pagination at some points.

Note:

**Number of returned objects**  Zammad has hard limits for the maximum returned objects. You can’t raise these limits.

**Number of total to return objects**  Zammad does not provide a total count of objects available for your query. This forces you to cycle through the pages until Zammad no longer returns further objects.

In order to use pagination you’ll need two get options: `per_page` and `page`. Combine them like so to receive 5 results from the first result page: `?page=1&per_page=5` - count page up to get more results.

17.7 Sorting search results

Zammad allows you to sort your search results by field if needed.

- **sort_by**  Append `?sort_by={row name}` to your query to sort by a specific row that appears in the search result.
- **order_by**  Append `?order_by={direction}` to your query to switch in between ascending and descending order.
  
  Directions are: `asc` and `desc`.

Note:  Usually you’ll want to combine both parameters in your searches - e.g.: `?query={search string}&sort_by={row name}&order_by={direction}`

17.8 Actions on behalf of other users

Note:  The user used for running the on behalf query requires `admin.user` permission.

Running API queries on behalf of other users allows you to e.g. create tickets on behalf of the user. The UI will display these kind of operations much better.

To do so, add a new HTTP header named `X-On-Behalf-Of` to your request. The value of this header can be one of the following:

- user ID
- user login
- user email

On behalf of is available for all endpoints.
17.9 Encoding

The API expects UTF-8 encoding. Keep in mind that especially when using URLs with get options (e.g. ?query=this) you may need to encode your URL accordingly.

If you want to learn more about URL encoding, this Wikipedia article may be of help.
Note: To see or not to see

Please note that below samples were provided with admin and ticket.agent permissions. Some attributes / information may not be available in specific situations.

Please see our Permission Guide to get better insights.

18.1 me - current user

Required permission: any

GET Request sent: /api/v1/users/me

Response:

```json
{
    "id": 3,
    "organization_id": 2,
    "login": "chris@chrispresso.com",
    "firstname": "Christopher",
    "lastname": "Miller",
    "email": "chris@chrispresso.com",
    "image": "7a6a0d1d94ad2037153cf3a6c1b49a53",
    "image_source": null,
    "web": "",
    "phone": "",
    "fax": "",
    "mobile": "",
    "department": null,
    "street": "",
    "zip": "",
    "city": "",
    "country": "",
    "address": null,
    "vip": false,
    "verified": false,
    "active": true,
}
```

(continues on next page)
"note": "",
"last_login": "2021-11-03T12:26:53.410Z",
"source": null,
"login_failed": 0,
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_office_replacement_id": null,
"preferences": {
    "notification_config": {
        "matrix": {
            "create": {
                "criteria": {
                    "owned_by_me": true,
                    "owned_by_nobody": true,
                    "subscribed": true,
                    "no": false
                },
                "channel": {
                    "email": true,
                    "online": true
                }
            },
            "update": {
                "criteria": {
                    "owned_by_me": true,
                    "owned_by_nobody": true,
                    "subscribed": true,
                    "no": false
                },
                "channel": {
                    "email": true,
                    "online": true
                }
            }
        },
        "reminder_reached": {
            "criteria": {
                "owned_by_me": true,
                "owned_by_nobody": false,
                "subscribed": false,
                "no": false
            }
        }
    }
}


```
},
  "channel":
  {
    "email": true,
    "online": true
  }
},
  "escalation":
  {
    "criteria":
    {
      "owned_by_me": true,
      "owned_by_nobody": false,
      "subscribed": false,
      "no": false
    },
    "channel":
    {
      "email": true,
      "online": true
    }
  }
},
  "locale": "en-us",
  "intro": true
},
"updated_by_id": 3,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:15.975Z",
"updated_at": "2021-11-03T12:26:55.642Z",
"role_ids":
[
  1,
  2
],
"organization_ids":
[],
"authorization_ids":
[],
"karma_user_ids":
[],
"group_ids":
{
  "1":
  [
    "full"
  ],
  "2":
  [
    "full"
  ]
},
```

(continues on next page)
18.2 List

Required permission: ticket.agent or admin.user

**Note:** Technically any, listings will return users own information only.

GET-Request sent: /api/v1/users

Response:

```json
HTTP/1.1 200 OK

[{
  "id": 1,
  "organization_id": null,
  "login": "-",
  "firstname": "-",
  "lastname": "-",
  "email": "",
  "image": null,
  "image_source": null,
  "web": "",
  "phone": "",
  "fax": "",
  "mobile": "",
  "department": "",
  "street": "",
  "zip": "",
  "city": "",
  "country": "",
  "address": "",
  "vip": false,
  "verified": false,
  "active": false,
  "note": "",
  "last_login": null,
  "source": null,
  "login_failed": 0,
  "out_of_office": false,
  "out_of_office_start_at": null,
  "out_of_office_end_at": null,
}]
```
"out_of_office_replacement_id": null,
"preferences":
{
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:51:12.786Z",
"updated_at": "2021-11-03T11:51:12.786Z",
"role_ids":
[],
"organization_ids":
[],
"authorization_ids":
[],
"karma_user_ids":
[],
"group_ids":
{}
},
{
"id": 2,
"organization_id": 1,
"login": "nicole.braun@zammad.org",
"firstname": "Nicole",
"lastname": "Braun",
"email": "nicole.braun@zammad.org",
"image": null,
"image_source": null,
"web": "",
"phone": "",
"fax": "",
"mobile": "",
"department": "",
"street": "",
"zip": "",
"city": "",
"country": "",
"address": "",
"vip": false,
"verified": false,
"active": true,
"note": "",
"last_login": null,
"source": null,
"login_failed": 0,
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_office_replacement_id": null,
"preferences":
{
"tickets_closed": 0,
"tickets_open": 1
}


```json
}

},

"updated_by_id": 2,
"created_by_id": 1,
"created_at": "2021-11-03T11:51:13.703Z",
"updated_at": "2021-11-03T12:01:05.411Z",
"role_ids":
[
  3
],
"organization_ids":
[],
"authorization_ids":
[],
"karma_user_ids":
[],
"group_ids":
{}
},

{

"id": 3,
"organization_id": 2,
"login": "chris@chrispresso.com",
"firstname": "Christopher",
"lastname": "Miller",
"email": "chris@chrispresso.com",
"image": "7a6a0d1d94ad2037153cf3a6c1b49a53",
"image_source": null,
"web": "",
"phone": "",
"fax": "",
"mobile": "",
"department": null,
"street": "",
"zip": "",
"city": "",
"country": "",
"address": null,
"vip": false,
"verified": false,
"active": true,
"note": "",
"last_login": "2021-11-03T12:26:53.410Z",
"source": null,
"login_failed": 0,
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_officeReplacement_id": null,
"preferences":
{

"notification_config":
{

(continues on next page)
"matrix":
{
  "create":
  {
    "criteria":
    {
      "owned_by_me": true,
      "owned_by_nobody": true,
      "subscribed": true,
      "no": false
    },
    "channel":
    {
      "email": true,
      "online": true
    }
  },
  "update":
  {
    "criteria":
    {
      "owned_by_me": true,
      "owned_by_nobody": true,
      "subscribed": true,
      "no": false
    },
    "channel":
    {
      "email": true,
      "online": true
    }
  },
  "reminder_reached":
  {
    "criteria":
    {
      "owned_by_me": true,
      "owned_by_nobody": false,
      "subscribed": false,
      "no": false
    },
    "channel":
    {
      "email": true,
      "online": true
    }
  },
  "escalation":
  {
    "criteria":
    {
      "owned_by_me": true,
"owned_by_nobody": false,
"subscribed": false,
"no": false
},
"channel":
{
"email": true,
"online": true
}
}
},
"locale": "en-us",
"intro": true
},
"updated_by_id": 3,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:15.975Z",
"updated_at": "2021-11-03T12:26:55.642Z",
"role_ids":
[
  1,
  2
],
"organization_ids": []
,"authorization_ids": []
,"karma_user_ids": []
,"group_ids":
{
"1":
[
  "full"
],
"2":
[
  "full"
],
"3":
[
  "full"
]
},
"id": 4,
"organization_id": 2,
"login": "jacob@chrispresso.com",
"firstname": "Jacob",
"lastname": "Smith"
"email": "jacob@chrispresso.com",
"image": "95afc1244af5cb8b77edcd7224c5d5f8",
"image_source": null,
"web": "",
"phone": "",
"fax": "",
"mobile": "",
"department": null,
"street": "",
"zip": "",
"city": "",
"country": "",
"address": null,
"vip": false,
"verified": false,
"active": true,
"note": "",
"last_login": null,
"source": null,
"login_failed": 0,
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_office_replacement_id": null,
"preferences": {
  "notification_config": {
    "matrix": {
      "create": {
        "criteria": {
          "owned_by_me": true,
          "owned_by_nobody": true,
          "subscribed": true,
          "no": false
        },
        "channel": {
          "email": true,
          "online": true
        }
      },
      "update": {
        "criteria": {
          "owned_by_me": true,
          "owned_by_nobody": true,
          "subscribed": true,
          "no": false
        }
      }
    }
  }
}
"no": false,
},
"channel":
{
  "email": true,
  "online": true
}
},
"reminder_reached":
{
  "criteria":
  {
    "owned_by_me": true,
    "owned_by_nobody": false,
    "subscribed": false,
    "no": false
  },
  "channel":
  {
    "email": true,
    "online": true
  }
},
"escalation":
{
  "criteria":
  {
    "owned_by_me": true,
    "owned_by_nobody": false,
    "subscribed": false,
    "no": false
  },
  "channel":
  {
    "email": true,
    "online": true
  }
}
},
"locale": "en-us"
},
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:16.160Z",
"updated_at": "2021-11-03T11:57:16.214Z",
"role_ids":
[ 1, 2 ],
"organization_ids":

(continues on next page)
empty, 
"authorization_ids": 
empty, 
"karma_user_ids": 
empty, 
"group_ids": 
{
  "1": 
  [ 
    "full"
  ],
  "2": 
  [ 
    "full"
  ],
  "3": 
  [ 
    "full"
  ]
}
}, 
{
  "id": 5,
  "organization_id": 2,
  "login": "emma@chrispresso.com",
  "firstname": "Emma",
  "lastname": "Taylor",
  "email": "emma@chrispresso.com",
  "image": "b64fef91c29105b4a08a2a69be08eda3",
  "image_source": null,
  "web": "",
  "phone": "",
  "fax": "",
  "mobile": "",
  "department": null,
  "street": "",
  "zip": "",
  "city": "",
  "country": "",
  "address": null,
  "vip": false,
  "verified": false,
  "active": true,
  "note": "",
  "last_login": null,
  "source": null,
  "login_failed": 0,
  "out_of_office": false,
  "out_of_office_start_at": null,
  "out_of_office_end_at": null,
  "out_of_office_replacement_id": null,
  "preferences":
}
```json
{
    "notification_config": {
        "matrix": {
            "create": {
                "criteria": {
                    "owned_by_me": true,
                    "owned_by_nobody": true,
                    "subscribed": true,
                    "no": false
                },
                "channel": {
                    "email": true,
                    "online": true
                }
            },
            "update": {
                "criteria": {
                    "owned_by_me": true,
                    "owned_by_nobody": true,
                    "subscribed": true,
                    "no": false
                },
                "channel": {
                    "email": true,
                    "online": true
                }
            },
            "reminder_reached": {
                "criteria": {
                    "owned_by_me": true,
                    "owned_by_nobody": false,
                    "subscribed": false,
                    "no": false
                },
                "channel": {
                    "email": true,
                    "online": true
                }
            }
        }
    }
}
```
"criteria": {
"owned_by_me": true,
"owned_by_nobody": false,
"subscribed": false,
"no": false
},
"channel": {
"email": true,
"online": true
},

"locale": "en-us",
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:16.349Z",
"updated_at": "2021-11-03T11:57:16.409Z",
"role_ids": [2],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": {
"1": [],
"2": [],
"3": []
}
}
"lastname": "Lopez",
"email": "anna@example.com",
"image": "4b1cb1fae2e608ffa72099774e1f57ad",
"image_source": null,
"web": "",
"phone": "415-123-5858",
"fax": "",
"mobile": "",
"department": null,
"street": "",
"zip": "",
"city": "",
"country": "",
"address": "Golden Gate Bridge\nSan Francisco, CA 94129",
"vip": false,
"verified": false,
"active": true,
"note": "likes espresso romano - recommended espresso con panna",
"last_login": null,
"source": null,
"login_failed": 0,
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_office_replacement_id": null,
"preferences": {},
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:16.526Z",
"updated_at": "2021-11-03T11:57:16.611Z",
"role_ids": [
  3
],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": {}},
{
  "id": 7,
  "organization_id": 3,
  "login": "samuel@example.com",
  "firstname": "Samuel",
  "lastname": "Lee",
  "email": "samuel@example.com",
  "image": "5911d228f3588c36a72d80eb0c1e4d08",}
"image_source": null,
"web": "",
"phone": "855-666-7777",
"fax": "",
"mobile": "",
"department": null,
"street": "",
"zip": "",
"city": "",
"country": "",
"address": "5201 Blue Lagoon Drive\n8th Floor & 9th Floor\nMiami, FL 33126",
"vip": false,
"verified": false,
"active": true,
"note": "likes americano, did order two units",
"last_login": null,
"source": null,
"login_failed": 0,
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_officeReplacement_id": null,
"preferences": {},
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:16.748Z",
"updated_at": "2021-11-03T11:57:16.861Z",
"role_ids":
[ 3
],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": []
},
{
"id": 8,
"organization_id": 3,
"login": "emily@example.com",
"firstname": "Emily",
"lastname": "Adams",
"email": "emily@example.com",
"image": "99ba64a89f7783c099c304c9b00ff9e8",
"image_source": null,
"web": "",
"phone": "0061 2 1234 7777",
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_officeReplacement_id": null,
"preferences": {},
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:16.748Z",
"updated_at": "2021-11-03T11:57:16.861Z",
"role_ids":
[ 3
],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": []
}
"fax": "",
"mobile": "",
"department": null,
"street": "",
"zip": "",
"city": "",
"country": "",
"address": "Bennelong Point\nSydney NSW 2000",
"vip": false,
"verified": false,
"active": true,
"note": "did order café au lait, ask next time if the flavor was as expected",
"last_login": null,
"source": null,
"login_failed": 0,
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_office_replacement_id": null,
"preferences": {},
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:17.000Z",
"updated_at": "2021-11-03T11:57:17.060Z",
"role_ids": [
  3
],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": {}
}
{
"id": 9,
"organization_id": 4,
"login": "ryan@example.com",
"firstname": "Ryan",
"lastname": "Parker",
"email": "ryan@example.com",
"image": "0e405c60b5deb780feb7ebebd37ff5e0",
"image_source": null,
"web": "",
"phone": "0049 30 1234 5678",
"fax": "",
"mobile": "",
"department": null,
"street": "", "zip": "", "city": "", "country": "", "address": "Brandenburger Tor 7\n10117 Berlin", "vip": false, "verified": false, "active": true, "note": "no latte but macchiato", "last_login": null, "source": null, "login_failed": 0, "out_of_office": false, "out_of_office_start_at": null, "out_of_office_end_at": null, "out_of_office_replacement_id": null, "preferences": {}, "updated_by_id": 1, "created_by_id": 1, "created_at": "2021-11-03T11:57:17.190Z", "updated_at": "2021-11-03T11:57:17.250Z", "role_ids": [3], "organization_ids": [], "authorization_ids": [], "karma_user_ids": [], "group_ids": {}},
{
  "id": 10, "organization_id": null, "login": "david@example.com", "firstname": "David", "lastname": "Bell", "email": "david@example.com", "image": "d829d234f377f231534802df6d5500a7", "image_source": null, "web": "", "phone": "0033 892 12 34 56", "fax": "", "mobile": "", "department": null, "street": "", "zip": "", "city": "", (continues on next page)
"country": "",
"address": "Eiffel Tower\n5 Avenue Anatole France\n75007 Paris",
"vip": false,
"verified": false,
"active": true,
"note": "did order viennese melange, ask next time if the flavor was as expected",
"last_login": null,
"source": null,
"login_failed": 0,
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_office_replacement_id": null,
"preferences": {
},
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:17.495Z",
"updated_at": "2021-11-03T11:57:17.561Z",
"role_ids": [
  3
],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": {}},
{
"id": 11,
"organization_id": null,
"login": "olivia@example.com",
"firstname": "Olivia",
"lastname": "Ross",
"email": "olivia@example.com",
"image": "b6f7a2d56544bb471eb3a3c238c7d964",
"image_source": null,
"web": "",
"phone": "0044 20 1234 5678",
"fax": "",
"mobile": "",
"department": null,
"street": "",
"zip": "",
"city": "",
"country": "",
"address": "Westminster\nLondon SW1A 0AA",
"vip": false,
"verified": false,
"active": true,
"note": "",
"last_login": null,
"source": null,
"login_failed": 0,
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_office_replacement_id": null,
"preferences": {},
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:17.741Z",
"updated_at": "2021-11-03T11:57:17.794Z",
"role_ids": [
  3
],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": {}
}

18.3 Search

Required permission: ticket.agent or admin.user

GET-Request sent: /api/v1/users/search?query=organization.name:{search string}&limit=10

Tip: ▼ Sort and Order ▲

Have a look into our Sorting search results Section.

Response:

```
HTTP/1.1 200 OK

[
  {
    "id": 8,
    "organization_id": 3,
    "login": "emily@example.com",
  }
]```
"firstname": "Emily",
"lastname": "Adams",
"email": "emily@example.com",
"image": "99ba64a89f7783c099c304c9b00ff9e8",
"image_source": null,
"web": "",
"phone": "0061 2 1234 7777",
"fax": "",
"mobile": "",
"department": null,
"street": "",
"zip": "",
"city": "",
"country": "",
"address": "Bennelong Point
Sydney NSW 2000",
"vip": false,
"verified": false,
"active": true,
"note": "did order café au lait, ask next time if the flavor was as expected",
"last_login": null,
"source": null,
"login_failed": 0,
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_office_replacement_id": null,
"preferences": {}
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:17.000Z",
"updated_at": "2021-11-03T11:57:17.060Z",
"role_ids": [
3
],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": {}
},
{
"id": 7,
"organization_id": 3,
"login": "samuel@example.com",
"firstname": "Samuel",
"lastname": "Lee",
"email": "samuel@example.com",
"image": "5911d228f3588c36a72d80eb0c1e4d08",
"image_source": null,
"web": "",
"phone": "855-666-7777",
"fax": "",
"mobile": "",
"vip": false,
"verified": false,
"department": null,
"street": "",
"zip": "",
"city": "",
"country": "",
"address": "5201 Blue Lagoon Drive\n 8th Floor & 9th Floor\n Miami, FL 33126",
"vip": false,
"verified": false,
"active": true,
"note": "likes americano, did order two units",
"last_login": null,
"source": null,
"login_failed": 0,
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_office_replacement_id": null,
"preferences": {},
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:16.748Z",
"updated_at": "2021-11-03T11:57:16.861Z",
"role_ids": [3],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": {}
},
{
"id": 6,
"organization_id": 3,
"login": "anna@example.com",
"firstname": "Anna",
"lastname": "Lopez",
"email": "anna@example.com",
"image": "4b1cb1f2e608ffaf72099ff741f57ad",
"image_source": null,
"web": "",
"phone": "415-123-5858",
"fax": "",
"mobile": "",
"department": null,
"street": "",
"zip": "",
"city": "",
"country": "",
"address": "Golden Gate Bridge\n San Francisco, CA 94129",
"vip": false,
"verified": false,
"active": true,
"note": ""}
"note": "likes espresso romano - recommended espresso con panna",
"last_login": null,
"source": null,
"login_failed": 0,
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_office Replacement_id": null,
"preferences": {},
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:16.526Z",
"updated_at": "2021-11-03T11:57:16.611Z",
"role_ids": [
3
],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": {}
]

18.4 Show

Required permission: ticket.agent or admin.user or ticket.customer (shared organization)

Note: Technically any, listings will return users own information only.

GET-Request sent: /api/v1/users/{id}
Response:

```
HTTP/1.1 200 OK

{
  "id": 11,
  "organization_id": null,
  "login": "olivia@example.com",
  "firstname": "Olivia",
  "lastname": "Ross",
  "email": "olivia@example.com",
  "image": "b6f7a2d56544bb471eb3a3c238c7d964",
  "image_source": null,
  "web": ",",
  "phone": "0044 20 1234 5678",
  "fax": ",",
  "mobile": ",",
  "department": null,
```
18.5 Create

Required permission: admin.user or ticket.agent

---

**Note:** This depends on permissions

Agents can't set user passwords, roles or group permission. Instead Zammad will apply to default sign up role.

Technically unauthenticated user creation is possible if you manage to provide the required CSRF token (out of scope of this documentation). If you don’t want that, consider disabling user registration.

---

**Tip:** Creation payloads can be big

Unsure which attributes you can use or set? Run a get query on any fitting user existing in your instance already.

---

POST-Request sent: /api/v1/users
{  
  "firstname": "Jane",
  "lastname": "Doe",
  "email": "jdoe@example.com",
  "login": "jdoe",
  "organization": "Sample Corp.",
  "roles": [
    "Agent",
    "Customer"
  ]
}

Response:

HTTP Code 201 Created

{  
  "id": 16,
  "organization_id": 5,
  "login": "jdoe",
  "firstname": "Jane",
  "lastname": "Doe",
  "email": "jdoe@example.com",
  "image": null,
  "image_source": null,
  "web": 
  "phone": 
  "fax": 
  "mobile": 
  "department": null,
  "street": 
  "zip": 
  "city": 
  "country": 
  "address": null,
  "vip": false,
  "verified": false,
  "active": true,
  "note": 
  "last_login": null,
  "source": null,
  "login_failed": 0,
  "out_of_office": false,
  "out_of_office_start_at": null,
  "out_of_office_end_at": null,
  "out_of_office_replacement_id": null,
  "preferences": {
    "notification_config": {
      "matrix": {
        "create": {
          "criteria": {
            "owned_by_me": true,
            "owned_by_nobody": true,
          }
        }
      }
    }
  }
}
"subscribed": true,
"no": false
},
"channel": {
"email": true,
"online": true
}
},
"update": {
"criteria": {
"owned_by_me": true,
"owned_by_nobody": true,
"subscribed": true,
"no": false
},
"channel": {
"email": true,
"online": true
}
},
"reminder_reached": {
"criteria": {
"owned_by_me": true,
"owned_by_nobody": false,
"subscribed": false,
"no": false
},
"channel": {
"email": true,
"online": true
}
},
"escalation": {
"criteria": {
"owned_by_me": true,
"owned_by_nobody": false,
"subscribed": false,
"no": false
},
"channel": {
"email": true,
"online": true
}
}
},
"locale": "en-us"
"role_ids": [2, 3],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": {}
}

18.6 Update

Required permission: admin.user or ticket.agent

Note: This depends on permissions
Agents can't set user passwords, roles or group permission. Instead Zammad will apply to default sign up role.

PUT-Request sent: /api/v1/users/{id}

```json
{
  "phone": "+49 30 55 57 160 00",
  "department": "Sales",
  "address": "Marienstr. 18\n10117 Berlin"
}
```

Response:

```
# HTTP/1.1 200 OK

{
  "id": 16,
  "organization_id": 5,
  "login": "jdoe",
  "firstname": "Jane",
  "lastname": "Doe",
  "email": "jdoe@example.com",
  "image": null,
  "image_source": null,
  "web": ",",
  "phone": "+49 30 55 57 160 00",
  "fax": ",",
  "mobile": ",",
  "department": "Sales",
  "street": ",",
  "zip": ",",
  "city": ",",
  "country": ",",
  "address": "Marienstr. 18\n10117 Berlin",
```

(continues on next page)
"vip": false,
"verified": false,
"active": true,
"note": "",
"last_login": null,
"source": null,
"login_failed": 0,
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_office_replacement_id": null,
"preferences": {
    "notification_config": {
        "matrix": {
            "create": {
                "criteria": {
                    "owned_by_me": true,
                    "owned_by_nobody": true,
                    "subscribed": true,
                    "no": false
                },
                "channel": {
                    "email": true,
                    "online": true
                }
            },
            "update": {
                "criteria": {
                    "owned_by_me": true,
                    "owned_by_nobody": true,
                    "subscribed": true,
                    "no": false
                },
                "channel": {
                    "email": true,
                    "online": true
                }
            },
            "reminder_reached": {
                "criteria": {
                    "owned_by_me": true,
                    "owned_by_nobody": false,
                    "subscribed": false,
                    "no": false
                },
                "channel": {
                    "email": true,
                    "online": true
                }
            },
            "escalation": {
                "criteria": {
            
        
    
}
"owned_by_me": true,
"owned_by_nobody": false,
"subscribed": false,
"no": false
},
"channel": {
"email": true,
"online": true
}
}

"locale": "en-us"
},
"updated_by_id": 3,
"created_by_id": 3,
"created_at": "2021-11-03T14:42:36.855Z",
"updated_at": "2021-11-03T14:49:20.018Z",
"role_ids": [2, 3],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": {}
}

18.7 Delete

Required permission: admin.user

Danger: This is a permanent removal

Please note that removing users cannot be undone. Zammad will also remove references - thus potentially tickets!
Removing users with references in e.g. activity streams is not possible via API - this will be indicated by "error": "Can't delete, object has references.". This is not a bug.
Consider using Data Privacy via UI for more control instead.

DELETE-Request sent: /api/v1/users/{id}
Response:

```
HTTP/1.1 200 OK

{}
```
19.1 List

Required permission: ticket.agent or admin.organization

Note: Technically any, customers can only see their own organization if applicable.

GET-Request sent: /api/v1/organizations
Response:

```json
HTTP/1.1 200 OK

[
  {
    "id": 1,
    "name": "Zammad Foundation",
    "shared": true,
    "domain": "",
    "domain_assignment": false,
    "active": true,
    "note": "",
    "updated_by_id": 1,
    "created_by_id": 1,
    "created_at": "2021-11-03T11:51:13.683Z",
    "updated_at": "2021-11-03T11:51:13.822Z",
    "member_ids": [
      2
    ]
  },
  {
    "id": 2,
    "name": "Chrispresso Inc.",
    "shared": true,
    "domain": "",
    "domain_assignment": false,
    "active": true,
    "note": "Manufacturer of individual coffee products.",
    "updated_by_id": 1,
    "created_by_id": 1,
  }
]
```

(continues on next page)
"created_at": "2021-11-03T11:57:15.817Z",
"updated_at": "2021-11-03T11:57:15.817Z",
"member_ids": [
   5,
   4,
   3
]
},
{
   "id": 3,
   "name": "Awesome Customer Inc.",
   "shared": true,
   "domain": "",
   "domain_assignment": false,
   "active": true,
   "note": "Global distributor of communication and security products, electrical and
   electronic wire & cable."
   "updated_by_id": 1,
   "created_by_id": 1,
   "created_at": "2021-11-03T11:57:15.825Z",
   "updated_at": "2021-11-03T11:57:15.825Z",
   "member_ids": [
      8,
      7,
      6
   ]
}
{
   "id": 4,
   "name": "Good Customer Inc.",
   "shared": true,
   "domain": "",
   "domain_assignment": false,
   "active": true,
   "note": "Search the world's information, including webpages, images, videos and
   more. Good Customer has many special features to help you find exactly what you're
   looking for."
   "updated_by_id": 1,
   "created_by_id": 1,
   "created_at": "2021-11-03T11:57:15.839Z",
   "updated_at": "2021-11-03T11:57:15.839Z",
   "member_ids": [
      9
   ]
}
{
   "id": 5,
   "name": "Test",
   "shared": true,
   "domain": "",
   "domain_assignment": false,
   "active": true,
19.2 Search

Required permission: ticket.agent or admin.organization

GET-Request sent: /api/v1/organizations/search?query=inc&limit=10

Tip: ↓ Sort and Order ↑
Have a look into our Sorting search results Section.

Response:

```
HTTP/1.1 200 OK
{
  
  "id": 4,
  "name": "Good Customer Inc.",
  "shared": true,
  "domain": "",
  "domain_assignment": false,
  "active": true,
  "note": "Search the world's information, including webpages, images, videos and more. Good Customer has many special features to help you find exactly what you're looking for.",
  "updated_by_id": 1,
  "created_by_id": 1,
  "created_at": "2021-11-03T11:57:15.839Z",
  "updated_at": "2021-11-03T11:57:15.839Z",
  "member_ids": [
    9
  ],
}
{
  
  "id": 3,
  "name": "Awesome Customer Inc.",
  "shared": true,
  "domain": "",
  "domain_assignment": false,
  "active": true,
  "note": "Global distributor of communication and security products, electrical and electronic wire & cable.",
  "updated_by_id": 3,
  "created_by_id": 3,
  "created_at": "2021-11-03T14:28:28.555Z",
  "updated_at": "2021-11-03T15:04:03.149Z",
  "member_ids": []
}
```

(continues on next page)
19.3 Show

Required permission: ticket.agent or admin.organization

Note: Technically any - users in question can only see their own organization.
"note": "Manufacturer of individual coffee products.",
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:15.817Z",
"updated_at": "2021-11-03T11:57:15.817Z",
"member_ids": [5, 4, 3]
}

### 19.4 Create

Required permission: admin.organization

**POST**-Request sent: /api/v1/organizations

```json
{
    "name": "Sample Corp.",
    "shared": false,
    "domain": "example.com",
    "domain_assignment": true,
    "active": true,
    "note": "Just a sample, aint that nice?",
    "members": [  
        "olivia@example.com",
        "jdoe",
        "david@example.com"
    ]
}
```

Response:

```
HTTP/1.1 201 Created

{
    "id": 6,
    "name": "Sample Corp.",
    "shared": false,
    "domain": "example.com",
    "domain_assignment": true,
    "active": true,
    "note": "Just a sample, aint that nice?",
    "updated_by_id": 3,
    "created_by_id": 3,
    "created_at": "2021-11-03T17:38:39.527Z",
    "updated_at": "2021-11-03T17:38:39.768Z",
    "member_ids": [10, 16, 17]
}
```

(continues on next page)
19.5 Update

Required permission: admin.organization

PUT-Request sent: /api/v1/organizations/{id}

```
{
    "name": "Sample Corp.",
    "shared": false,
    "domain": "",
    "domain_assignment": false,
    "active": true,
    "note": "This was a triumph - I'm making a note here - H-U-G-E success!",
    "members": [
        "olivia@example.com",
        "david@example.com"
    ]
}
```

Response:

```
HTTP code 200 OK
{
    "id": 6,
    "name": "Sample Corp.",
    "shared": false,
    "domain": "",
    "domain_assignment": false,
    "active": true,
    "note": "This was a triumph - I'm making a note here - H-U-G-E success!",
    "updated_by_id": 3,
    "created_by_id": 3,
    "created_at": "2021-11-03T17:38:39.527Z",
    "updated_at": "2021-11-03T17:40:59.740Z",
    "member_ids": [
        11,
        10
    ]
}
```
19.6 Delete

Required permission: admin.organization

**Danger:** This is a permanent removal

Please note that removing organizations cannot be undone.

Removing organizations with references in e.g. activity streams or users is not possible via API - this will be indicated by "error": "Can't delete, object has references.". This is not a bug.

Consider using Data Privacy via UI for more control instead.

DELETE-Request sent: /api/v1/organization/{id}

Response:

```
HTTP CODE 200 OK
{}
```
Note: Confusing setting for endpoint
Please note that follow_up_possible may not work as expected. The possible values are yes or new_ticket!

20.1 List

Required permission: admin.group

GET-Request sent: /api/v1/groups

Response:

```
# HTTP Code 200 OK

[
    {
        "id": 1,
        "signature_id": 1,
        "email_address_id": null,
        "name": "Sales",
        "assignment_timeout": null,
        "follow_up_possible": "yes",
        "follow_up_assignment": true,
        "active": true,
        "note": "Standard Group/Pool for Tickets.",
        "updated_by_id": 1,
        "created_by_id": 1,
        "created_at": "2021-11-03T11:51:13.449Z",
        "updated_at": "2021-11-03T11:57:16.357Z",
        "user_ids": [3, 4, 5]
    },
    {
        "id": 2,
        "signature_id": null,
    }
]
```
20.2 Show

Required permission: admin.group

GET-Request sent: /api/v1/groups/{id}

Response:

```
HTTP/1.1 200 OK

{
  "id": 2,
}
```
20.3 Create

Required permission: admin.group

POST-Request sent: /api/v1/groups

```json
{
    "name": "Amazing Group",
    "signature_id": 1,
    "email_address_id": 1,
    "assignment_timeout": 180,
    "follow_up_possible": "new_ticket",
    "follow_up_assignment": false,
    "active": true,
    "note": "Look at my group, my group is amazing!"
}
```

Response:

```
HTTP/1.1 201 Created
```

```json
{
    "id": 7,
    "signature_id": 1,
    "email_address_id": 3,
    "name": "Amazing Group",
    "assignment_timeout": 180,
    "follow_up_possible": "new_ticket",
    "follow_up_assignment": false,
    "active": true,
    "note": "Look at my group, my group is amazing!",
    "updated_by_id": 3,
}
```
20.4 Update

Required permission: admin.group

PUT-Request sent: /api/v1/groups/{id}

```
{  
    "name": "Amazing Group",
    "signature_id": 1,
    "email_address_id": 3,
    "assignment_timeout": 0,
    "follow_up_possible": "new_ticket",
    "follow_up_assignment": true,
    "active": true,
    "note": "Look at my group, my group is amazing!"
}
```

Response:

```
HTTP/1.1 200 OK

{
    "id": 7,
    "signature_id": 1,
    "email_address_id": 3,
    "name": "Amazing Group",
    "assignment_timeout": 0,
    "follow_up_possible": "new_ticket",
    "follow_up_assignment": true,
    "active": true,
    "note": "Look at my group, my group is amazing!",
    "updated_by_id": 3,
    "created_by_id": 3,
    "created_at": "2021-11-08T13:09:41.526Z",
    "updated_at": "2021-11-08T13:36:24.571Z",
    "user_ids": []
}
```
20.5 Delete

Required permission: admin.group

Danger: This is a permanent removal

Please note that removing groups cannot be undone.

Removing organizations with references in e.g. activity streams or tickets is not possible via API - this will be indicated by "error": "Can't delete, object has references.". This is not a bug.

Consider setting affected groups to inactive instead or ensure to move all existing tickets to new groups.

DELETE-Request sent: /api/v1/groups/{id}

Response:

HTTP Code 200 OK

{}
Note: Calendars belong to Zammads SLA calculation.

21.1 List

Required permission: admin.calendar

GET-Request sent: /api/v1/calendars

```
status: 200
[
  {
    "id": 2,
    "name": "Test calendar",
    "timezone": "Europe/Berlin",
    "business_hours": {
      "mon": {
        "active": true,
        "timeframes": [
          "09:00",
          "17:00"
        ]
      },
      "tue": {
        "active": true,
        "timeframes": [
          "09:00",
          "17:00"
        ]
      },
      "wed": {
        "active": true,
        "timeframes": [
          "09:00",
          "17:00"
        ]
      }
    }
  }
]
```

(continues on next page)
[ "09:00",
 "17:00"
 ]

"thu":{
 "active":true,
 "timeframes":[
 [ "09:00",
 "17:00"
 ]
 }

"fri":{
 "active":true,
 "timeframes":[
 [ "09:00",
 "17:00"
 ]
 }

"sat":{
 "active":false,
 "timeframes":[
 [ "10:00",
 "14:00"
 ]
 }

"sun":{
 "active":false,
 "timeframes":[
 [ "10:00",
 "14:00"
 ]
 }
 },
"default":false,
"ical_url":"
,"public_holidays":{
 "2021-11-10":{
 "active":true,
 "summary":"Feast day 1"
 },
 "2021-11-11":{
 "active":true,
"summary":"Feast day 2"
}
,"last_log":null,
"last_sync":"2021-11-10T13:14:20.835Z",
"updated_by_id":3,
"created_by_id":3,
"created_at":"2021-11-10T13:14:20.835Z",
"updated_at":"2021-11-10T13:14:20.835Z"
}
]

### 21.2 Show

Required permission: admin.calendar

GET-Request sent: /api/v1/calendars/{id}

**Status: 200**

```
{
  "id":2,
  "name":"Test calendar",
  "timezone":"Europe/Berlin",
  "business_hours":{
    "mon":{
      "active":true,
      "timeframes":[
        [09:00,17:00]
      ]
    },
    "tue":{
      "active":true,
      "timeframes":[
        [09:00,17:00]
      ]
    },
    "wed":{
      "active":true,
      "timeframes":[
        [09:00,17:00]
      ]
    }
  }
}
```
"thu": {
  "active": true,
  "timeframes": [
    ["09:00", "17:00"],
  ]
},
"fri": {
  "active": true,
  "timeframes": [
    ["09:00", "17:00"],
  ]
},
"sat": {
  "active": false,
  "timeframes": [
    ["10:00", "14:00"],
  ]
},
"sun": {
  "active": false,
  "timeframes": [
    ["10:00", "14:00"],
  ]
},
"default": false,
"ical_url": ",
"public_holidays": {
  "2021-11-10": {
    "active": true,
    "summary": "Feast day 1"
  },
  "2021-11-11": {
    "active": true,
    "summary": "Feast day 2"
  }
},
"last_log": null,
21.3 Create

Required permission: admin.calendar

POST-Request sent: /api/v1/calendars

```json
{
    "name": "Test calendar",
    "timezone": "Europe/Berlin",
    "business_hours": {
        "mon": {
            "active": true,
            "timeframes": [
                ["09:00", "17:00"]
            ]
        },
        "tue": {
            "active": true,
            "timeframes": [
                ["09:00", "17:00"]
            ]
        },
        "wed": {
            "active": true,
            "timeframes": [
                ["09:00", "17:00"]
            ]
        },
        "thu": {
            "active": true,
            "timeframes": [
                ["09:00", "17:00"]
            ]
        }
    }
}
```
"fri":{
    "active":true,
    "timeframes":[
        ["09:00","17:00"
    ]
},
"sat":{
    "active":false,
    "timeframes":[
        ["10:00","14:00"
    ]
},
"sun":{
    "active":false,
    "timeframes":[
        ["10:00","14:00"
    ]
}
"ical_url":"
,"public_holidays":{
    "2021-11-10":{
        "active":true,
        "summary":"Feast day 1"
    },
    "2021-11-11":{
        "active":true,
        "summary":"Feast day 2"
    }
},
"note":"
,"id":"c-1"}
"name":"Test calendar",
"timezone":"Europe/Berlin",
"business_hours":{
  "mon":{
    "active":true,
    "timeframes":[
      [
        "09:00",
        "17:00"
      ]
    ]
  },
  "tue":{
    "active":true,
    "timeframes":[
      [
        "09:00",
        "17:00"
      ]
    ]
  },
  "wed":{
    "active":true,
    "timeframes":[
      [
        "09:00",
        "17:00"
      ]
    ]
  },
  "thu":{
    "active":true,
    "timeframes":[
      [
        "09:00",
        "17:00"
      ]
    ]
  },
  "fri":{
    "active":true,
    "timeframes":[
      [
        "09:00",
        "17:00"
      ]
    ]
  },
  "sat":{
    "active":false,
    "timeframes":[
      [
        "09:00",
        "17:00"
      ]
    ]
  }
}
21.4 Update

Required permission: admin.calendar

PUT-Request sent: /api/v1/calendars/{id}

```json
{
    "name": "Test calendar Update",
    "timezone": "Europe/Berlin",
    "default": false,
    "business_hours": {
        "mon": {
            "active": true,
            "timeframes": [
                "10:00",
                "14:00"
            ]
        }
    }
}
```
"09:00",
"17:00"
]
}
},
"tue":{
"active":true,
"timeframes": [
    "09:00",
    "17:00"
]
}
},
"wed":{
"active":true,
"timeframes": [
    "09:00",
    "17:00"
]
}
},
"thu":{
"active":true,
"timeframes": [
    "09:00",
    "17:00"
]
}
},
"fri":{
"active":true,
"timeframes": [
    "09:00",
    "17:00"
]
}
},
"sat":{
"active":false,
"timeframes": [
    "10:00",
    "14:00"
]
}
},
"sun":{
"active":false,
"timeframes": [
  [
    "10:00",
    "14:00"
  ]
],
"ical_url": "",
"public_holidays": {
  "2021-11-10": {
    "active": true,
    "summary": "Feast day 1"
  },
  "2021-11-11": {
    "active": true,
    "summary": "Feast day 2"
  }
},
"note": "",
"id": 2
}

Response:

Status: 200 OK

{"id": 2,
"name": "Test calendar Update",
"timezone": "Europe/Berlin",
"default": false,
"ical_url": "",
"business_hours": {
  "mon": {
    "active": true,
    "timeframes": [
      [
        "09:00",
        "17:00"
      ]
    ]
  },
  "tue": {
    "active": true,
    "timeframes": [
      [
        "09:00",
        "17:00"
      ]
    ]
  }
}

(continues on next page)
"wed": {
   "active": true,
   "timeframes": [
      ["09:00",
       "17:00"
      ]
   ]
},
"thu": {
   "active": true,
   "timeframes": [
      ["09:00",
       "17:00"
      ]
   ]
},
"fri": {
   "active": true,
   "timeframes": [
      ["09:00",
       "17:00"
      ]
   ]
},
"sat": {
   "active": false,
   "timeframes": [
      ["10:00",
       "14:00"
      ]
   ]
},
"sun": {
   "active": false,
   "timeframes": [
      ["10:00",
       "14:00"
      ]
   ]
},
"public_holidays": {
   "2021-11-10": {
      "active": true,
      "summary": "Feast day 1"
   },
   "2021-11-11": {
      (continues on next page)
21.5 Delete

Required permission: admin.calendar

Danger:  This is a permanent removal

Please note that removing Calendar configurations cannot be undone.

Removing calendars with references in SLA configurations is not possible via API - this will be indicated by "error": "Can't delete, object has references.". This is not a bug.

DELETE-Request sent: /api/v1/calendars/{id}

Status: 200

{}
SERVICE-LEVEL AGREEMENTS (SLA)

Note: SLAs depend on Zammads Calendars.

22.1 List

Required permission: admin.sla

GET-Request sent: /api/v1/slas

Response:

```
HTTP/1.1 200 OK

[
    {
        "id": 2,
        "calendar_id": 1,
        "name": "new sla",
        "first_response_time": 120,
        "response_time": null,
        "update_time": 120,
        "solution_time": 120,
        "condition": {
            "ticket.state_id": {
                "operator": "is",
                "value": "2"
            }
        },
        "updated_by_id": 3,
        "created_by_id": 3,
        "created_at": "2021-11-10T12:54:39.368Z",
        "updated_at": "2021-11-10T12:54:39.368Z"
    }
]
```
22.2 Show

Required permission: admin.sla

GET-Request sent: /api/v1/slas/{id}

Response:

```json
HTTP Code 200 OK

{
  "id":2,
  "calendar_id":1,
  "name":"new sla",
  "first_response_time":120,
  "response_time":null,
  "update_time":120,
  "solution_time":120,
  "condition":{
    "ticket.state_id":{
      "operator":"is",
      "value":"2"
    }
  },
  "updated_by_id":3,
  "created_by_id":3,
  "created_at":"2021-11-10T12:54:39.368Z",
  "updated_at":"2021-11-10T12:54:39.368Z"
}
```

22.3 Create

Required permission: admin.sla

POST-Request sent: /api/v1/slas

Response:
22.4 Update

Required permission: admin.sla

PUT-Request sent: /api/v1/slas/[id]

```json
{
  "name": "update sla",
  "first_response_time": "120",
  "response_time": "",
  "update_time": "120",
  "solution_time": "120",
  "condition": {
    "ticket.state_id": {
      "operator": "is",
      "value": "2"
    }
  },
  "calendar_id": "1",
  "id": 2
}
```

Response:

```json
# HTTP Code 200 OK
{
  "id": 2,
  "calendar_id": 1,
}
```

(continues on next page)
"name":"update sla",
"first_response_time":120,
"response_time":null,
"update_time":120,
"solution_time":120,
"condition":{
   "ticket.state_id":{
      "operator":"is",
      "value":"2"
   }
},
"updated_by_id":3,
"created_by_id":3,
"created_at":"2021-11-10T12:54:39.368Z",
"updated_at":"2021-11-10T13:02:52.053Z"
}

22.5 Delete

Required permission: admin.sla

Danger: This is a permanent removal
Please note that removing SLA configurations cannot be undone.

DELETE-Request sent: /api/v1/slas/{id}
Response:

```
HTTP/1.1 200 OK
{}
```
Zammad comes with many ticket related endpoints. For better overview the grouped those up:

**23.1 Tickets**

**Warning:** Ticket endpoints depend on group permissions if the user you’re using is an agent. Because of this tickets may or may not be available.

### 23.1.1 List

Required permission: `ticket.agent` or `ticket.customer`

GET-Request sent: `/api/v1/tickets`

Response:

```
HTTP/1.1 200 OK
```

```json
[
{
    "id": 1,
    "group_id": 1,
    "priority_id": 2,
    "state_id": 1,
    "organization_id": 1,
    "number": "22001",
    "title": "Welcome to Zammad!",
    "owner_id": 1,
    "customer_id": 2,
    "note": null,
    "first_response_at": null,
    "first_response_escalation_at": null,
    "first_response_in_min": null,
    "first_response_diff_in_min": null,
    "close_at": null,
    "close_escalation_at": null,
    "close_in_min": null,
    "close_diff_in_min": null,
}
```
"update_escalation_at": null,
"update_in_min": null,
"update_diff_in_min": null,
"last_contact_at": "2021-11-03T11:51:13.790Z",
"last_contact_agent_at": null,
"last_contact_customer_at": "2021-11-03T11:51:13.790Z",
"last_owner_update_at": null,
"create_article_type_id": 5,
"create_article_sender_id": 2,
"article_count": 1,
"escalation_at": null,
"pending_time": null,
"type": null,
"time_unit": null,
"preferences": {},
"updated_by_id": 2,
"created_by_id": 2,
"created_at": "2021-11-03T11:51:13.759Z",
"updated_at": "2021-11-03T11:51:13.809Z"
},
{
  "id": 2,
  "group_id": 1,
  "priority_id": 2,
  "state_id": 4,
  "organization_id": 3,
  "number": "22002",
  "title": "Order 777555",
  "owner_id": 3,
  "customer_id": 6,
  "note": null,
  "first_response_at": null,
  "first_response_escalation_at": null,
  "first_response_in_min": null,
  "first_response_diff_in_min": null,
  "close_at": null,
  "close_escalation_at": null,
  "close_in_min": null,
  "close_diff_in_min": null,
  "update_escalation_at": null,
  "update_in_min": null,
  "update_diff_in_min": null,
  "last_contact_at": "2021-05-04T16:57:17.920Z",
  "last_contact_agent_at": "2021-05-03T10:57:17.904Z",
  "last_contact_customer_at": "2021-05-04T16:57:17.920Z",
  "last_owner_update_at": null,
  "create_article_type_id": 1,
  "create_article_sender_id": 2,
  "article_count": 3,
  "escalation_at": null,
  "pending_time": null,
  "type": null,
23.1.2 Search

Required permission: ticket.agent or ticket.customer

GET-Request sent: /api/v1/tickets/search?query={search string}&limit=10

Tip: ↓ Sort and Order ↑

Have a look into our Sorting search results Section.

Response:

```
HTTP/1.1 200 OK

{
    "tickets": [
        9,
        10,
        11
    ],
    "tickets_count": 3,
    "assets": {
        "Ticket": {
            "9": {
                "id": 9,
                "group_id": 1,
                "priority_id": 3,
                "state_id": 2,
                "organization_id": 7,
                "number": "22009",
                "title": "Need more information!",
                "owner_id": 5,
                "customer_id": 10,
                "note": null,
                "first_response_at": null,
                "first_response_escalation_at": null,
                "first_response_in_min": null,
                "first_response_diff_in_min": null,
                "close_at": null,
                "close_escalation_at": null,
```

(continues on next page)
"close_in_min": null,
"close_diff_in_min": null,
"update_escalation_at": null,
"update_in_min": null,
"update_diff_in_min": null,
"last_contact_at": "2021-11-03T05:42:19.141Z",
"last_contact_agent_at": "2021-11-03T05:42:19.141Z",
"last_contact_customer_at": "2021-11-03T02:57:19.141Z",
"last_owner_update_at": null,
"create_article_type_id": 1,
"create_article_sender_id": 2,
"article_count": 4,
"escalation_at": null,
"pending_time": null,
"type": null,
"time_unit": null,
"preferences": {},
"updated_by_id": 3,
"created_by_id": 10,
"created_at": "2021-11-03T02:57:19.141Z",
"updated_at": "2021-11-03T17:48:52.849Z",
"article_ids": [
  19,
  18,
  17,
  16
],
"ticket_time_accounting_ids": []
},
"10": {
  "id": 10,
  "group_id": 1,
  "priority_id": 3,
  "state_id": 1,
  "organization_id": 7,
  "number": "22010",
  "title": "Heads up!",
  "owner_id": 1,
  "customer_id": 11,
  "note": null,
  "first_response_at": null,
  "first_response_escalation_at": null,
  "first_response_in_min": null,
  "first_response_diff_in_min": null,
  "close_at": null,
  "close_escalation_at": null,
  "close_in_min": null,
  "close_diff_in_min": null,
  "update_escalation_at": null,
  "update_in_min": null,
  "update_diff_in_min": null,
  "last_contact_at": "2021-11-03T11:57:19.227Z"
"last_contact_agent_at": null,
"last_contact_customer_at": "2021-11-03T11:57:19.227Z",
"last_owner_update_at": null,
"create_article_type_id": 1,
"create_article_sender_id": 2,
"article_count": 1,
"escalation_at": null,
"pending_time": null,
"type": null,
"time_unit": null,
"preferences": {},
"updated_by_id": 3,
"created_by_id": 11,
"created_at": "2021-11-03T02:57:19.216Z",
"updated_at": "2021-11-03T17:48:52.730Z",
"article_ids": [20],
"ticket_time_accounting_ids": []
},
"11": {
  "id": 11,
  "group_id": 1,
  "priority_id": 3,
  "state_id": 1,
  "organization_id": 3,
  "number": "22011",
  "title": "Surprise - well done",
  "owner_id": 1,
  "customer_id": 6,
  "note": null,
  "first_response_at": null,
  "first_response_escalation_at": null,
  "first_response_in_min": null,
  "first_response_diff_in_min": null,
  "close_at": null,
  "close_escalation_at": null,
  "close_in_min": null,
  "close_diff_in_min": null,
  "update_escalation_at": null,
  "update_in_min": null,
  "update_diff_in_min": null,
  "last_contact_at": "2021-11-03T11:57:19.243Z",
  "last_contact_agent_at": null,
  "last_contact_customer_at": "2021-11-03T02:57:19.243Z",
  "last_owner_update_at": null,
  "create_article_type_id": 11,
  "create_article_sender_id": 2,
  "article_count": 1,
  "escalation_at": null,
  "pending_time": null,
  "type": null,
"time_unit": null,
"preferences": {},
"updated_by_id": 6,
"created_by_id": 6,
"created_at": "2021-11-03T02:57:19.243Z",
"updated_at": "2021-11-03T11:57:19.263Z",
"article_ids": [
  21
],
"ticket_time_accounting_ids": []
},
"User": {
"10": {
  "id": 10,
  "organization_id": null,
  "login": "david@example.com",
  "firstname": "David",
  "lastname": "Bell",
  "email": "david@example.com",
  "image": "d829d234f377f231534802df6d5500a7",
  "image_source": null,
  "web": "",
  "phone": "0033 892 12 34 56",
  "fax": "",
  "mobile": "",
  "department": "",
  "street": "",
  "zip": "",
  "city": "",
  "country": "",
  "address": "Eiffel Tower\n5 Avenue Anatole France\n75007 Paris",
  "vip": false,
  "verified": false,
  "active": true,
  "note": "did order viennese melange, ask next time if the flavor was as expected",
  "last_login": null,
  "source": null,
  "login_failed": 0,
  "out_of_office": false,
  "out_of_office_start_at": null,
  "out_of_office_end_at": null,
  "out_of_office_replacement_id": null,
  "preferences": {
    "tickets_closed": 1,
    "tickets_open": 3,
    "mail_delivery_failed": true,
    "mail_delivery_failed_data": "2021-11-08T13:38:32.059Z"
  },
  "updated_by_id": 1,
}
"created_by_id": 1,
"created_at": "2021-11-03T11:57:17.495Z",
"updated_at": "2021-11-08T13:45:04.107Z",
"role_ids": [
  3
],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": {},
"accounts": {}},
"1": {
  "id": 1,
  "organization_id": null,
  "login": "-",
  "firstname": "-",
  "lastname": "",
  "email": "",
  "image": null,
  "image_source": null,
  "web": "",
  "phone": "",
  "fax": "",
  "mobile": "",
  "department": "",
  "street": "",
  "zip": "",
  "city": "",
  "country": "",
  "address": "",
  "vip": false,
  "verified": false,
  "active": false,
  "note": "",
  "last_login": null,
  "source": null,
  "login_failed": 0,
  "out_of_office": false,
  "out_of_office_start_at": null,
  "out_of_office_end_at": null,
  "out_of_office_replacement_id": null,
  "preferences": {},
  "updated_by_id": 1,
  "created_by_id": 1,
  "created_at": "2021-11-03T11:51:12.786Z",
  "updated_at": "2021-11-03T11:51:12.786Z",
  "role_ids": [],
  "organization_ids": [],
  "authorization_ids": [],
  "karma_user_ids": [],
  "group_ids": {}}
"accounts": {},
},
"3": {
  "id": 3,
  "organization_id": 2,
  "login": "chris@chrispresso.com",
  "firstname": "Christopher",
  "lastname": "Miller",
  "email": "chris@chrispresso.com",
  "image": "7a6a0d1d94ad2037153cf3a6c1b49a53",
  "image_source": null,
  "web": "",
  "phone": "",
  "fax": "",
  "mobile": "",
  "department": "",
  "street": "",
  "zip": "",
  "city": "",
  "country": "",
  "address": "",
  "vip": false,
  "verified": false,
  "active": true,
  "note": "",
  "last_login": "2021-11-03T12:26:53.410Z",
  "source": null,
  "login_failed": 0,
  "out_of_office": false,
  "out_of_office_start_at": null,
  "out_of_office_end_at": null,
  "out_of_office_replacement_id": null,
  "preferences": {
    "notification_config": {
      "matrix": {
        "create": {
          "criteria": {
            "owned_by_me": true,
            "owned_by_nobody": true,
            "subscribed": true,
            "no": false
          },
          "channel": {
            "email": true,
            "online": true
          }
        },
        "update": {
          "criteria": {
            "owned_by_me": true,
            "owned_by_nobody": true,
            "subscribed": true,
            "no": false
          }
        }
      }
    }
  }
}
", "no": false },
  "channel": {
    "email": true,
    "online": true
  }
},
  "reminder_reached": {
    "criteria": {
      "owned_by_me": true,
      "owned_by_nobody": false,
      "subscribed": false,
      "no": false
    },
    "channel": {
      "email": true,
      "online": true
    }
  },
  "escalation": {
    "criteria": {
      "owned_by_me": true,
      "owned_by_nobody": false,
      "subscribed": false,
      "no": false
    },
    "channel": {
      "email": true,
      "online": true
    }
  }
},
"locale": "en-us",
"intro": true,
"chat": {
  "active": {"1": "on"
  }
},
"updated_by_id": 3,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:15.975Z",
"updated_at": "2021-11-08T13:45:07.798Z",
"role_ids": [1, 2],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [23.1. Tickets 205
1
],
"group_ids": {
  "1": [
    "full"
  ],
  "2": [
    "full"
  ],
  "3": [
    "full"
  ]
},
"accounts": {}
},
"4": {
  "id": 4,
  "organization_id": 2,
  "login": "jacob@chrispresso.com",
  "firstname": "Jacob",
  "lastname": "Smith",
  "email": "jacob@chrispresso.com",
  "image": "95afc124af5cb8b77edcd7224c5d5f8",
  "image_source": null,
  "web": "",
  "phone": "",
  "fax": "",
  "mobile": "",
  "department": null,
  "street": "",
  "zip": "",
  "city": "",
  "country": "",
  "address": null,
  "vip": false,
  "verified": false,
  "active": true,
  "note": "",
  "last_login": null,
  "source": null,
  "login_failed": 0,
  "out_of_office": false,
  "out_of_office_start_at": null,
  "out_of_office_end_at": null,
  "out_of_office_replacement_id": null,
  "preferences": {
    "notification_config": {
      "matrix": {
        "create": {
          "criteria": {
            "owned_by_me": true,
            "owned_by_nobody": true,
          }
        }
      }
    }
  }
}
"subscribed": true,
"no": false
},
"channel": {
"email": true,
"online": true
}
},
"update": {
"criteria": {
"owned_by_me": true,
"owned_by_nobody": true,
"subscribed": true,
"no": false
},
"channel": {
"email": true,
"online": true
}
},
"reminder_reached": {
"criteria": {
"owned_by_me": true,
"owned_by_nobody": false,
"subscribed": false,
"no": false
},
"channel": {
"email": true,
"online": true
}
},
"escalation": {
"criteria": {
"owned_by_me": true,
"owned_by_nobody": false,
"subscribed": false,
"no": false
},
"channel": {
"email": true,
"online": true
}
}

"locale": "en-us"
},
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:16.160Z",
"updated_at": "2021-11-03T11:57:16.214Z",
(continues on next page)
"role_ids": [
  1,
  2
],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": {
  "1": [
    "full"
  ],
  "2": [
    "full"
  ],
  "3": [
    "full"
  ]
},
"accounts": {}
},
"5": {
  "id": 5,
  "organization_id": 7,
  "login": "emma@chrispresso.com",
  "firstname": "Emma",
  "lastname": "Taylor",
  "email": "emma@chrispresso.com",
  "image": "b64fef91c29105b4a08a2a69be08eda3",
  "image_source": null,
  "web": "",
  "phone": "",
  "fax": "",
  "mobile": "",
  "department": null,
  "street": "",
  "zip": "",
  "city": "",
  "country": "",
  "address": null,
  "vip": false,
  "verified": false,
  "active": true,
  "note": "",
  "last_login": null,
  "source": null,
  "login_failed": 0,
  "out_of_office": false,
  "out_of_office_start_at": null,
  "out_of_office_end_at": null,
  "out_of_office_replacement_id": null,
  "preferences": {
    "notification_config": {
      (continues on next page)
"matrix": {
  "create": {
    "criteria": {
      "owned_by_me": true,
      "owned_by_nobody": true,
      "subscribed": true,
      "no": false
    },
    "channel": {
      "email": true,
      "online": true
    }
  },
  "update": {
    "criteria": {
      "owned_by_me": true,
      "owned_by_nobody": true,
      "subscribed": true,
      "no": false
    },
    "channel": {
      "email": true,
      "online": true
    }
  },
  "reminder_reached": {
    "criteria": {
      "owned_by_me": true,
      "owned_by_nobody": false,
      "subscribed": false,
      "no": false
    },
    "channel": {
      "email": true,
      "online": true
    }
  },
  "escalation": {
    "criteria": {
      "owned_by_me": true,
      "owned_by_nobody": false,
      "subscribed": false,
      "no": false
    },
    "channel": {
      "email": true,
      "online": true
    }
  }
},
"locale": "en-us"
},
  "updated_by_id": 3,
  "created_by_id": 1,
  "created_at": "2021-11-03T11:57:16.349Z",
  "updated_at": "2021-11-08T13:22:38.130Z",
  "role_ids": [
    2
  ],
  "organization_ids": [],
  "authorization_ids": [],
  "karma_user_ids": [],
  "group_ids": {
    "1": [
      "full"
    ],
    "2": [
      "full"
    ],
    "3": [
      "full"
    ]
  },
  "accounts": {}
},
"11": {
  "id": 11,
  "organization_id": 7,
  "login": "olivia@example.com",
  "firstname": "Olivia",
  "lastname": "Ross",
  "email": "olivia@example.com",
  "image": "b6f7a2d56544bb471eb3a3c238c7d964",
  "image_source": null,
  "web": "",
  "phone": "0044 20 1234 5678",
  "fax": "",
  "mobile": "",
  "department": "",
  "street": "",
  "zip": "",
  "city": "",
  "country": "",
  "address": "Westminster\r\nLondon SW1A 0AA",
  "vip": false,
  "verified": false,
  "active": true,
  "note": "",
  "last_login": null,
  "source": null,
  "login_failed": 0,
  "out_of_office": false,
  "out_of_office_start_at": null,
(continues on next page)
"out_of_office_end_at": null,
"out_of_office_replacement_id": null,
"preferences": {
    "tickets_closed": 0,
    "tickets_open": 1
},
"updated_by_id": 3,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:17.741Z",
"updated_at": "2021-11-03T17:48:52.739Z",
"role_ids": [
    3
],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": {},
"accounts": {}
},
"16": {
    "id": 16,
    "organization_id": 7,
    "login": "jdoe",
    "firstname": "Jane",
    "lastname": "Doe",
    "email": "jdoe@example.com",
    "image": null,
    "image_source": null,
    "web": "",
    "phone": "+49 30 55 57 160 00",
    "fax": "",
    "mobile": "",
    "department": "Sales",
    "street": "",
    "zip": "",
    "city": "",
    "country": "",
    "address": "Marienstr. 18\n10117 Berlin",
    "vip": false,
    "verified": false,
    "active": true,
    "note": "",
    "last_login": null,
    "source": null,
    "login_failed": 0,
    "out_of_office": false,
    "out_of_office_start_at": null,
    "out_of_office_end_at": null,
    "out_of_office_replacement_id": null,
    "preferences": {
        "notification_config": {
            "matrix": {
            (continues on next page)
"create": {
    "criteria": {
        "owned_by_me": true,
        "owned_by_nobody": true,
        "subscribed": true,
        "no": false
    },
    "channel": {
        "email": true,
        "online": true
    }
},
"update": {
    "criteria": {
        "owned_by_me": true,
        "owned_by_nobody": true,
        "subscribed": true,
        "no": false
    },
    "channel": {
        "email": true,
        "online": true
    }
},
"reminder_reached": {
    "criteria": {
        "owned_by_me": true,
        "owned_by_nobody": false,
        "subscribed": false,
        "no": false
    },
    "channel": {
        "email": true,
        "online": true
    }
},
"escalation": {
    "criteria": {
        "owned_by_me": true,
        "owned_by_nobody": false,
        "subscribed": false,
        "no": false
    },
    "channel": {
        "email": true,
        "online": true
    }
},
"locale": "en-us"
"updated_by_id": 3,
"created_by_id": 3,
"created_at": "2021-11-03T14:42:36.855Z",
"updated_at": "2021-11-08T13:20:18.500Z",
"role_ids": [
  2,
  3
],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": {},
"accounts": {}},
"6": {
"id": 6,
"organization_id": 3,
"login": "anna@example.com",
"firstname": "Anna",
"lastname": "Lopez",
"email": "anna@example.com",
"image": "4b1cb1fae2e608ffa72099774e1f57ad",
"image_source": null,
"web": "",
"phone": "415-123-5858",
"fax": "",
"mobile": "",
"department": null,
"street": "",
"zip": "",
"city": "",
"country": "",
"address": "Golden Gate Bridge\nSan Francisco, CA 94129",
"vip": false,
"verified": false,
"active": true,
"note": "likes espresso romano - recommended espresso con panna",
"last_login": null,
"source": null,
"login_failed": 0,
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_office_replacement_id": null,
"preferences": {},
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:16.526Z",
"updated_at": "2021-11-03T11:57:16.611Z",
"role_ids": [3]
],
(continues on next page)
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": {},
"accounts": {}
},
"7": {
"id": 7,
"organization_id": 3,
"login": "samuel@example.com",
"firstname": "Samuel",
"lastname": "Lee",
"email": "samuel@example.com",
"image": "5911d228f3588c36a72d80eb0c1e4d08",
"image_source": null,
"web": "",
"phone": "855-666-7777",
"fax": "",
"mobile": "",
"department": null,
"street": "",
"zip": "",
"city": "",
"country": "",
"address": "5201 Blue Lagoon Drive
8th Floor & 9th Floor
Miami, FL 33126",
"vip": false,
"verified": false,
"active": true,
"note": "likes americano, did order two units",
"last_login": null,
"source": null,
"login_failed": 0,
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_office_replacement_id": null,
"preferences": {},
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:16.748Z",
"updated_at": "2021-11-03T11:57:16.861Z",
"role_ids": [3]
],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": [],
"accounts": {}
},
"8": {
(continues on next page)
"id": 8,
"organization_id": 3,
"login": "emily@example.com",
"firstname": "Emily",
"lastname": "Adams",
"email": "emily@example.com",
"image": "99ba64a89f7783c099c304c9b00ff9e8",
"image_source": null,
"web": "",
"phone": "0061 2 1234 7777",
"fax": "",
"mobile": "",
"department": null,
"street": "",
"zip": "",
"city": "",
"country": "",
"address": "Bennelong Point\nSydney NSW 2000",
"vip": false,
"verified": false,
"active": true,
"note": "did order café au lait, ask next time if the flavor was as expected",
"last_login": null,
"source": null,
"login_failed": 0,
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_office_replacement_id": null,
"preferences": {},
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:17.000Z",
"updated_at": "2021-11-03T11:57:17.060Z",
"role_ids": [
3
],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": {},
"accounts": {}
"note": "People who create Tickets ask for help."
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:51:12.856Z",
"updated_at": "2021-11-08T13:38:31.573Z",
"permission_ids": [
  42,
  45,
  46,
  48,
  54
],
"group_ids": {}},
"1": {
"id": 1,
"name": "Admin",
"preferences": {},
"default_at_signup": false,
"active": true,
"note": "To configure your system."
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:51:12.831Z",
"updated_at": "2021-11-03T11:51:12.831Z",
"permission_ids": [
  1,
  41,
  51,
  61
],
"group_ids": {}},
"2": {
"id": 2,
"name": "Agent",
"preferences": {},
"default_at_signup": false,
"active": true,
"note": "To work on Tickets."
"updated_by_id": 3,
"created_by_id": 1,
"created_at": "2021-11-03T11:51:12.848Z",
"updated_at": "2021-11-03T14:42:36.875Z",
"permission_ids": [
  41,
  53,
  56,
  58,
  62
],
"group_ids": {}}


```
},

"Group": {
  "1": {
    "id": 1,
    "signature_id": 1,
    "email_address_id": null,
    "name": "Sales",
    "assignment_timeout": null,
    "follow_up_possible": "yes",
    "follow_up_assignment": true,
    "active": true,
    "note": "Standard Group/Pool for Tickets.",
    "updated_by_id": 1,
    "created_by_id": 1,
    "created_at": "2021-11-03T11:51:13.449Z",
    "updated_at": "2021-11-08T13:37:57.093Z",
    "user_ids": [
      4,
      5,
      3
    ]
  },
  "2": {
    "id": 2,
    "signature_id": null,
    "email_address_id": null,
    "name": "2nd Level",
    "assignment_timeout": null,
    "follow_up_possible": "yes",
    "follow_up_assignment": true,
    "active": true,
    "note": null,
    "updated_by_id": 1,
    "created_by_id": 1,
    "created_at": "2021-11-03T11:51:13.449Z",
    "updated_at": "2021-11-08T13:37:57.093Z",
    "user_ids": [
      4,
      5,
      3
    ]
  },
  "3": {
    "id": 3,
    "signature_id": null,
    "email_address_id": null,
    "name": "Service Desk",
    "assignment_timeout": null,
    "follow_up_possible": "yes",
    "follow_up_assignment": true,
    "active": true,
    "note": null,
    "updated_by_id": 1,
    "created_by_id": 1,
    "created_at": "2021-11-03T11:51:13.449Z",
    "updated_at": "2021-11-08T13:37:57.093Z",
    "user_ids": [
      4,
      5,
      3
    ]
  }
}
```

(continues on next page)
"note": null,
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:15.807Z",
"updated_at": "2021-11-08T13:57.102Z",
"user_ids": [
4,
5,
3
]
},
"Organization": {
"2": {
"id": 2,
"name": "Chrispresso Inc.",
"shared": true,
"domain": "",
"domain_assignment": false,
"active": true,
"note": "Manufacturer of individual coffee products.",
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:15.817Z",
"updated_at": "2021-11-08T13:22:38.145Z",
"member_ids": [
3,
4
]
},
"7": {
"id": 7,
"name": "Sample Corp.",
"shared": false,
"domain": "",
"domain_assignment": false,
"active": true,
"note": "This was a triumph - I'm making a note here - H-U-G-E success!",
"updated_by_id": 3,
"created_by_id": 3,
"created_at": "2021-11-03T17:05:52.613Z",
"updated_at": "2021-11-08T13:22:38.148Z",
"member_ids": [
5,
11,
16
]
},
"3": {
"id": 3,
"name": "Awesome Customer Inc.",
(continues on next page)
"shared": true,
"domain": ",
"domain_assignment": false,
"active": true,
"note": "Global distributor of communication and security products, electrical and electronic wire & cable."
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:57:15.825Z",
"updated_at": "2021-11-03T11:57:15.825Z",
"member_ids": [6, 7, 8]
}
}
}

Warning: tickets_count returns the current number of returned tickets, not the total amount.

23.1.3 Show

Required permission: ticket.agent or ticket.customer

GET-Request sent: /api/v1/tickets/{ticket id}

Response:

```json
HTTP/1.1 200 OK

{
  "id": 3,
  "group_id": 1,
  "priority_id": 2,
  "state_id": 4,
  "organization_id": 3,
  "number": "22003",
  "title": "Order 787556",
  "owner_id": 3,
  "customer_id": 7,
  "note": null,
  "first_response_at": null,
  "first_response_escalation_at": null,
  "first_response_in_min": null,
  "first_response_diff_in_min": null,
  "close_at": null,
  "close_escalation_at": null,
  "close_in_min": null,
}
```

(continues on next page)
23.1.4 Create

Required permission: `ticket.agent` or `ticket.customer`

**Tip:** On behalf of users

If you want to create tickets on behalf other users, use the `customer_id` attribute. `ticket.agent` is mandatory for this. Use `guess:{email address}` to save an API call if you don’t know the users ID or want to create the user in question.

**Add mention subscription right away**

Add the mentions attribute to your ticket payload and provide an array of user ids to directly subscribe them during ticket creation.

E.g.: "mentions": [1, 5, 7, 8],

```
POST-Request sent: /api/v1/tickets

{
    "title": "Help me!",
    "group": "2nd Level",
    "customer": "david@example.com",
    "article": {
        "subject": "My subject",
        "body": "I am a message!",
        "type": "note",
        "internal": false
    }
}
```
Response:

```json
{  
  "id": 19,  
  "group_id": 2,  
  "priority_id": 2,  
  "state_id": 1,  
  "organization_id": null,  
  "number": "22019",  
  "title": "Help me!",  
  "owner_id": 1,  
  "customer_id": 10,  
  "note": null,  
  "first_response_at": null,  
  "first_response_escalation_at": null,  
  "first_response_in_min": null,  
  "close_at": null,  
  "close_escalation_at": null,  
  "close_in_min": null,  
  "close_diff_in_min": null,  
  "update escalate_at": null,  
  "update_in_min": null,  
  "update_diff_in_min": null,  
  "last_contact_at": null,  
  "last_contact_agent_at": null,  
  "last_contact_customer_at": null,  
  "last_owner_update_at": null,  
  "create article_type_id": 10,  
  "create article_sender_id": 1,  
  "article_count": 1,  
  "escalation_at": null,  
  "pending_time": null,  
  "type": null,  
  "time_unit": null,  
  "preferences": {},  
  "updated_by_id": 3,  
  "created_by_id": 3,  
  "create at": "2021-11-08T14:17:41.913Z",  
  "updated_at": "2021-11-08T14:17:41.994Z",  
  "article_ids": [  
    30  
  ],  
  "ticket_time_accounting_ids": []
}
```

**Hint:** For more article attributes and options have a look into *Articles.*
23.1.5 Update

Required permission: ticket.agent or ticket.customer

PUT-Request sent: /api/v1/tickets/{ticket id}

```json
{
  "title": "No help for you",
  "group": "Sales",
  "state": "open",
  "priority": "3 high",
  "article": {
    "subject": "Update via API",
    "body": "Here's my reason for updating this ticket...",
    "internal": true
  }
}
```

**Note:** Above example provides an article. This article is a new article and does not affect any existing ones.

Response:

```
HTTP/1.1 200 OK

{
  "id": 19,
  "group_id": 1,
  "priority_id": 3,
  "state_id": 2,
  "organization_id": null,
  "number": "22019",
  "title": "No help for you",
  "owner_id": 1,
  "customer_id": 10,
  "note": null,
  "first_response_at": null,
  "first_response escalation at": null,
  "first_response_in_min": null,
  "first_response diff_in_min": null,
  "close_at": null,
  "close escalation at": null,
  "close_in_min": null,
  "close diff_in_min": null,
  "update escalation at": null,
  "update_in_min": null,
  "update diff_in_min": null,
  "last_contact at": null,
  "last_contact agent at": null,
  "last_contact customer at": null,
  "last_owner update at": null,
  "create article type id": 10,
  "create article sender id": 1,
```

(continues on next page)
Tip: Adding attachments
Attachment payloads are identical to the POST method, just use PUT instead.

23.1.6 Delete

Required permission: admin

Danger: This is a permanent removal
Please note that removing tickets cannot be undone. All data (e.g.: articles & attachments) will be lost.

DELETE-Request sent: /api/v1/tickets/{ticket id}
Response:

```
HTTP/1.1 200 OK
{}
```

23.2 Articles

23.2.1 General information about ticket articles

Some attributes of articles might not be straight forward or come with fairly many options - below list hopefully helps you on this journey.

- **content_type** Zammad supports text/html for HTML formatted text or text/plain for plain text. This allows you to have better formatting options if you need them.

  Hint: Zammad web UI usually uses text/html.
Zammad supports a huge number of article types. Below list may be incomplete depending on your instance and possibly installed add-ons / custom changes.

Note: If not stated otherwise, all article types below are communication articles and thus affecting SLA calculation in Zammad defaults.

What's the difference?! Communication articles provide the option to reply automatically. Which actions exactly are available depends on the article type and e.g. recipient lists.

**email** This allows you to create incoming or outgoing email articles.

Hint: This allows you to send Email via API. However, this highly depends on the chosen sender.

**phone** Indicates phone notes.

**web** Usually used by customers only. This type is being used when ever your customer uses the web UI to create articles.

**note** When ever a communication does not fit (e.g.: internal notes) choose note. Zammad also uses this article type as default fall back.

Note: This is not a communication article.

**sms** This type is being used for Zammad’s SMS integration.

**chat** This article type is technically a place holder and is only available via API.

**fax** This article type is technically a place holder and is only available via API.

**twitter status & twitter direct-message** These articles types are used by Zammad’s twitter channel. Technically you can use these to automatically respond to existing requests via twitter.

**facebook feed post & facebook feed comment** These articles types are used by Zammad’s facebook channel. Technically you can use these to automatically respond to existing requests via facebook.

**telegram personal-message** Used by Zammad’s Telegram channel. Technically you can use these to automatically respond to existing requests via Telegram.

**internal** This attribute allows you to set the visibility of your articles. For internal visible only use true, for visibly for your customers as well use false.
Warning: Visibility: internal doesn’t mean it’s silent

If you set an article to internal: true but choose to send an email, please be aware that said Email is still being sent out!

sender Indicates which use did create the article. You can choose from:

- Agent
- Customer
- System

Warning: Depending of above selection, some article types may not be available or behave different. Please be aware that System causes users not being able to read the bodies (this works similar to Zammad’s trigger displaying in tickets).

List Articles by Ticket

Required permission: ticket.agent or ticket.customer

GET-Request sent: /api/v1/ticket_articles/by_ticket/{ticket id}

Response:

```
HTTP/1.1 200 OK

[{
   "id": 9,
   "ticket_id": 5,
   "type_id": 1,
   "sender_id": 2,
   "from": "David Bell <david@example.com>",
   "to": "order@chrispresso.com",
   "cc": null,
   "subject": null,
   "reply_to": null,
   "message_id": null,
   "message_id_md5": null,
   "in_reply_to": null,
   "content_type": "text/html",
   "references": null,
   "body": "Hi,<br/>please send me:<br/>
1 x Café Kopi susu<br/>
4 x Viennese melange<br/>
Delivery Address:<br/>
David Bell<br/>
Eiffel Tower<br/>
5 Avenue Anatole France<br/>
75007 Paris<br/>
David Bell",
   "internal": false,
   "preferences": {},
   "updated_by_id": 10,
   "created_by_id": 10,
   "origin_by_id": null,
   "created_at": "2021-08-02T11:57:18.068Z",
}]
```
"updated_at": "2021-08-02T11:57:18.068Z",
"attachments": [],
"type": "email",
"sender": "Customer",
"created_by": "david@example.com",
"updated_by": "david@example.com"
},
{
  "id": 10,
  "ticket_id": 5,
  "type_id": 1,
  "sender_id": 1,
  "from": "Emma Taylor via <order@chrispresso.com>",
  "to": "David Bell <david@example.com>",
  "cc": null,
  "subject": null,
  "reply_to": null,
  "message_id": null,
  "message_id_md5": null,
  "in_reply_to": null,
  "content_type": "text/html",
  "references": null,
  "body": "Hi David,<br>
<br>
nice, we will ship it to your delivery address:<br>
Eiffel Tower
5 Avenue Anatole France
75007 Paris.<br>
You will get it till Wednesday.<br>

Greetings,<br>
Emma Taylor",
  "internal": false,
  "preferences": {},
  "created_by_id": 5,
  "updated_by_id": 5,
  "origin_by_id": null,
  "created_at": "2021-08-03T09:57:18.121Z",
  "updated_at": "2021-08-03T09:57:18.121Z",
  "attachments": [],
  "type": "email",
  "sender": "Agent",
  "created_by": "emma@chrispresso.com",
  "updated_by": "emma@chrispresso.com"
}
]

List specific article

Required permission: ticket.agent or ticket.customer

GET-Request sent: /api/v1/ticket_articles/{article id}

Response:

```
HTTP/1.0 200 OK
{
(continues on next page)
"id": 9,
"ticket_id": 5,
"type_id": 1,
"sender_id": 2,
"from": "David Bell <david@example.com>",
"to": "order@chrispresso.com",
"cc": null,
"subject": null,
"reply_to": null,
"message_id": null,
"message_id_md5": null,
"in_reply_to": null,
"content_type": "text/html",
"references": null,
"body": "Hi,

Please send me:
1 x Café Kopi susu
→
4 x Viennese melange

Delivery Address:
David Bell

Eiffel Tower
5 Avenue Anatole France
75007 Paris

David Bell",
"internal": false,
"preferences": {},
"updated_by_id": 10,
"created_by_id": 10,
"origin_by_id": null,
"created_at": "2021-08-02T11:57:18.068Z",
"updated_at": "2021-08-02T11:57:18.068Z",
"attachments": [],
"type": "email",
"sender": "Customer",
"created_by": "david@example.com",
"updated_by": "david@example.com"
}

Create

Required permission: ticket.agent or ticket.customer

Tip: If you want to create articles on behalf other users (e.g. for a phone note), use the origin_by_id attribute. ticket.agent permission is mandatory for this.

POST-Request sent: /api/v1/ticket_articles

Plain article

Article with attached files

Article with inline images

{ "ticket_id": 5,
  "subject": "Call note",
  "body": "Called the customer and discussed their issues.<br/>Turns out these were caused by invalid configurations - solved."
}

(continues on next page)
"content_type": "text/html",
"type": "phone",
"internal": false,
"sender": "Agent",
"time_unit": "15"
}

Response:

```json
HTTP/1.1 201 Created
{
  "id": 33,
  "ticket_id": 5,
  "type_id": 5,
  "sender_id": 1,
  "from": "Christopher Miller",
  "to": null,
  "cc": null,
  "subject": "Call note",
  "reply_to": null,
  "message_id": null,
  "message_id_md5": null,
  "in_reply_to": null,
  "content_type": "text/html",
  "references": null,
  "body": "Called the customer and discussed their issues.<br>Turns out these were caused by invalid configurations - solved."
  "internal": false,
  "preferences": {},
  "updated_by_id": 3,
  "created_by_id": 3,
  "origin_by_id": null,
  "created_at": "2021-11-08T16:13:35.962Z",
  "updated_at": "2021-11-08T16:13:35.962Z",
  "attachments": [],
  "type": "phone",
  "sender": "Agent",
  "created_by": "chris@chrispresso.com",
  "updated_by": "chris@chrispresso.com"
}
```

**Hint:** The first attachment example does work, remove the second “generalized” part to try the payload out.

```json
{
  "ticket_id": 5,
  "to": "",
  "cc": "",
  "subject": "some subject",
}
```
"body": "Please see attached file...",
"content_type": "text/plain",
"type": "note",
"internal": true,
"time_unit": "25",
"attachments": [
  {
    "filename": "portal.txt",
    "data": "VGhlIGNha2UgaXMgYSBsaWU=",
    "mime-type": "text/plain"
  },
  {
    "filename": "{filename}",
    "data": "{file content base64 encoded}",
    "mime-type": "{attachments mime-type}"
  }
]
}

Response:

```
HTTP/1.1 201 Created

{
  "id": 35,
  "ticket_id": 5,
  "type_id": 10,
  "sender_id": 1,
  "from": "Christopher Miller",
  "to": "",
  "cc": "",
  "subject": "some subject",
  "reply_to": null,
  "message_id": null,
  "message_id_md5": null,
  "in_reply_to": null,
  "content_type": "text/plain",
  "references": null,
  "body": "Please see attached file...",
  "internal": true,
  "preferences": {},
  "updated_by_id": 3,
  "created_by_id": 3,
  "origin_by_id": null,
  "created_at": "2021-11-09T12:02:55.434Z",
  "updated_at": "2021-11-09T12:02:55.434Z",
  "attachments": [
    {
      "id": 17,
      "filename": "portal.txt",
      "size": "19",
      "preferences": {}
    }
  ]
}
```
"Mime-Type": "text/plain"
},
}
"type": "note",
"sender": "Agent",
"created_by": "chris@chrispresso.com",
"updated_by": "chris@chrispresso.com"
}

Inline images can be used by providing data URIs in your HTML markup.

{ "ticket_id": 5,
"to": "",
"cc": "some subject",
"body": "Let's see the <b>phoenix</b> "}

(continues from previous page)
"content_type": "text/html",
"type": "note",
"internal": false,
"time_unit": "12"
}

Response:

201 Created

{  
  "id": 37,
  "ticket_id": 5,
  "type_id": 10,
  "sender_id": 1,
  "from": "Christopher Miller",
  "to": "",
  "cc": "",
  "subject": "some subject",
  "reply_to": null,
  "message_id": null,
  "message_id_md5": null,
  "in_reply_to": null,
  "content_type": "text/html",
  "references": null,
  "body": "Let's see the <b>phoenix</b> <img src="/api/v1/ticket_attachment/→5/37/19?view=inline" style="max-width:100%;">",
  "internal": false,
  "preferences": {},
  "updated_by_id": 3,
  "created_by_id": 3,
  "origin_by_id": null,
  "created_at": "2021-11-09T12:10:49.375Z",
  "updated_at": "2021-11-09T12:10:49.375Z",
  "attachments": [
    {
      "id": 19,
      "filename": "image1.png",
      "size": "3735",
      "preferences": {
        "Content-Type": "image/png",
        "Mime-Type": "image/png",
        "Content-ID": "5.e384b84e-bfef-49f7-af22-8546fb99f8dc@fqdn",
        "Content-Disposition": "inline"
      }
    }
  ],
  "type": "note",
  "sender": "Agent",
  "created_by": "chris@chrispresso.com",
  "updated_by": "chris@chrispresso.com"
}
Receive attachments

Now that you have all those fancy attachments within your tickets, you may want to download specific ones.

GET-Request sent: /api/v1/ticket_attachment/{ticket id}/{article id}/{attachment id}

Response:

Hint: If you’re not sure which articles a ticket / article contains, please retrieve affected articles first.

23.3 Priorities

23.3.1 List

Required permission: admin.object or ticket.agent or ticket.customer

GET-Request sent: /api/v1/ticket_priorities

Response:

```json
[
  {
    "id": 1,
    "name": "1 low",
    "default_create": false,
    "ui_icon": "low-priority",
    "ui_color": "low-priority",
    "note": null,
    "active": true,
    "updated_by_id": 1,
    "created_by_id": 1,
    "created_at": "2021-11-03T11:51:13.559Z",
    "updated_at": "2021-11-03T11:51:13.572Z"
  },
  {
    "id": 2,
    "name": "2 normal",
    "default_create": true,
    "ui_icon": null,
    "ui_color": null
  }
]
```

(continues on next page)
23.3.2 Show

Required permission: admin.object or ticket.agent or ticket.customer

GET-Request sent: /api/v1/ticket_priorities/{id}

Response:

```
HTTP/1.1 200 OK

{
  "id": 3,
  "name": "3 high",
  "default_create": false,
  "ui_icon": "important",
  "ui_color": "high-priority",
  "note": null,
  "active": true,
  "updated_by_id": 1,
  "created_by_id": 1,
  "created_at": "2021-11-03T11:51:13.579Z",
  "updated_at": "2021-11-03T11:51:13.579Z"
}
```
23.3.3 Create

Required permission: admin.object

POST-Request sent: /api/v1/ticket_priorities

```
{
    "name": "4 disaster",
    "default_create": false,
    "ui_icon": "important",
    "ui_color": "high-priority",
    "note": "Added via API for disasterious situations."
}
```

Response:
```
HTTP/1.1 201 Created

{
    "id": 4,
    "name": "4 disaster",
    "default_create": false,
    "ui_icon": "important",
    "ui_color": "high-priority",
    "note": "Added via API for disasterious situations.",
    "active": true,
    "updated_by_id": 3,
    "created_by_id": 3,
    "created_at": "2021-11-08T15:31:57.704Z",
    "updated_at": "2021-11-08T15:31:57.704Z"
}
```

23.3.4 Update

Required permission: admin.object

PUT-Request sent: /api/v1/ticket_priorities/{id}

```
{
    "ui_icon": "",
    "ui_color": "",
    "note": "Adjusted via API - not so important"
}
```

Response:
```
HTTP/1.1 200 OK

{
    "id": 3,
    "ui_icon": "",
    "ui_color": "",
    "note": "Adjusted via API - not so important",
    "updated_by_id": 3,
    "created_at": "2021-11-08T15:31:57.704Z",
    "updated_at": "2021-11-08T15:31:57.704Z"
}
```
"name": "3 high",
"default_create": false,
"active": true,
"created_by_id": 1,
"created_at": "2021-11-03T11:51:13.579Z",
"updated_at": "2021-11-08T15:33:12.181Z"
}

23.3.5 Delete

Required permission: admin.object

Danger: This is a permanent removal

Please note that removing priorities cannot be undone. Removing ticket priorities with references in tickets is not possible via API - this will be indicated by "error": "Can't delete, object has references.". This is not a bug.

Consider either setting said priority to active: false or adjust all tickets with the to remove priority to another priority.

DELETE-Request sent: /api/v1/ticket_priorities/{id}
Response:

```
HTTP/1.1 200 OK
{}
```

23.4 States

Warning: Creating, changing or removing states via below endpoints require further steps via console. See Make new states available to UI

23.4.1 List

Required permission: admin.object or ticket.agent or ticket.customer

GET-Request sent: /api/v1/ticket_states
Response:

```
HTTP/1.1 200 OK
[
  {
    "id": 1,
```
"state_type_id": 1,
"name": "new",
"next_state_id": null,
"ignore_escalation": false,
"default_create": true,
"default_follow_up": false,
"note": null,
"active": true,
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:51:13.504Z",
"updated_at": "2021-11-03T11:51:13.520Z"
},
{
"id": 2,
"state_type_id": 2,
"name": "open",
"next_state_id": null,
"ignore_escalation": false,
"default_create": false,
"default_follow_up": true,
"note": null,
"active": true,
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:51:13.518Z",
"updated_at": "2021-11-03T11:51:13.518Z"
},
{
"id": 3,
"state_type_id": 3,
"name": "pending reminder",
"next_state_id": null,
"ignore_escalation": true,
"default_create": false,
"default_follow_up": false,
"note": null,
"active": true,
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-11-03T11:51:13.528Z",
"updated_at": "2021-11-03T11:51:13.528Z"
},
{
"id": 4,
"state_type_id": 5,
"name": "closed",
"next_state_id": null,
"ignore_escalation": true,
"default_create": false,
"default_follow_up": false,
"note": null,
<table>
<thead>
<tr>
<th>id</th>
<th>state_type_id</th>
<th>name</th>
<th>next_state_id</th>
<th>ignore_escalation</th>
<th>default_create</th>
<th>default_follow_up</th>
<th>note</th>
<th>active</th>
<th>updated_by_id</th>
<th>created_by_id</th>
<th>created_at</th>
<th>updated_at</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>4</td>
<td>pending close</td>
<td>4</td>
<td>true</td>
<td>false</td>
<td>false</td>
<td>null</td>
<td>true</td>
<td>1</td>
<td>1</td>
<td>2021-11-03T11:13:55Z</td>
<td>2021-11-03T11:13:55Z</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>merged</td>
<td>null</td>
<td>true</td>
<td>false</td>
<td>false</td>
<td>null</td>
<td>true</td>
<td>1</td>
<td>1</td>
<td>2021-11-03T11:13:54Z</td>
<td>2021-11-03T11:13:54Z</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>removed</td>
<td>null</td>
<td>true</td>
<td>false</td>
<td>false</td>
<td>null</td>
<td>false</td>
<td>1</td>
<td>1</td>
<td>2021-11-03T11:13:54Z</td>
<td>2021-11-03T11:13:54Z</td>
</tr>
</tbody>
</table>
23.4.2 Show

Required permission: admin.object or ticket.agent or ticket.customer

GET-Request sent: /api/v1/ticket_states/{id}

Response:

```
HTTP/1.1 200 OK

{
    "id": 4,
    "state_type_id": 5,
    "name": "closed",
    "next_state_id": null,
    "ignore_escalation": true,
    "default_create": false,
    "default_follow_up": false,
    "note": null,
    "active": true,
    "updated_by_id": 1,
    "created_by_id": 1,
    "created_at": "2021-11-03T11:51:13.535Z",
    "updated_at": "2021-11-03T11:51:13.535Z"
}
```

23.4.3 Create

Required permission: admin.object

Note: Below payload makes use of state_type_id which is a instance specific set of IDs. State types indicate how the state will work. As there’s no endpoint for retrieving these, please use the rails console.

POST-Request sent: /api/v1/ticket_states

```
{
    "name": "in progress",
    "state_type_id": 2,
    "ignore_escalation": true,
    "active": true
}
```

Response:

```
HTTP/1.1 201 Created

{
    "id": 8,
    "state_type_id": 2,
    "name": "in progress",
    "next_state_id": null,
    "ignore_escalation": true,
    "default_create": false,
    "default_follow_up": false,
    "note": null,
    "active": true,
    "updated_by_id": 1,
    "created_by_id": 1,
    "created_at": "2021-11-03T11:51:13.535Z",
    "updated_at": "2021-11-03T11:51:13.535Z"
}
```
23.4.4 Update

Required permission: admin.object

PUT-Request sent: /api/v1/ticket_states/{id}

```json
{
  "note": "State created & updated via API"
}
```

Response:

```
HTTP Code 200 OK

{
  "id": 8,
  "note": "State created & updated via API",
  "updated_by_id": 3,
  "name": "in progress",
  "state_type_id": 2,
  "next_state_id": null,
  "ignore_escalation": true,
  "default_create": false,
  "default_follow_up": false,
  "active": true,
  "created_by_id": 3,
  "created_at": "2021-11-08T15:08:21.671Z",
  "updated_at": "2021-11-08T15:08:21.671Z"
}
```

23.4.5 Delete

Required permission: admin.object

**Danger:** This is a permanent removal

Please note that removing ticket states cannot be undone.

Removing ticket states with references in tickets is not possible via API - this will be indicated by "error": "Can't delete, object has references.". This is not a bug.
Consider either setting said state to `active: false` or adjust all tickets with the to remove state to another state.

DELETE-Request sent: `/api/v1/ticket_states/{id}`
Response:

```
HTTP/1.1 200 OK
{}
```

23.5 Tags

23.5.1 Ticket scope

List

Required permission: `ticket.agent` or `admin.tag`

GET-Request sent: `/api/v1/tags?object=Ticket&o_id={ticket id}`

Sample response:

```
HTTP/1.1 200 OK
{
   "tags": [
      "americano",
      "complaint"
   ]
}
```

Search

Required permission: `ticket.agent` or `admin.tag`

GET-Request sent: `/api/v1/tag_search?term={tag name}`

**Hint:** Zammad will return all tags that contain your search phrase.

Sample response:

```
HTTP/1.1 200 OK
[
   {
      "id": 1,
      "value": "americano"
   },
   
   ...
]
```
Add

Required permission: ticket.agent or admin.tag

POST-Request sent: /api/v1/tags/add

```json
{
  "item": "{tag name}"
}
```

**Hint:** This will create the tag if it doesn’t exist and the user has permission to do so.

Response:

```
HTTP/1.1 201 Created
true
```

Remove

Required permission: ticket.agent or admin.tag

DELETE-Request sent: /api/v1/tags/remove

```json
{
  "item": "{tag name}"
}
```

Response:

```
HTTP/1.1 201 Created
true
```
23.5.2 Administration scope

Admin - List

Required permission: admin.tag
GET-Request sent: /api/v1/tag_list
Sample response:

```
HTTP/1.1 200 OK
```
```
[
  {
    "id": 1,
    "name": "americano",
    "count": 0
  },
  {
    "id": 2,
    "name": "complaint",
    "count": 0
  },
  {
    "id": 3,
    "name": "viennese melange",
    "count": 0
  }
]
```

Admin - Create

Required permission: admin.tag
POST-Request sent: /api/v1/tag_list

```
{
  "name": "tag 5"
}
```

Response:

```
HTTP/1.1 200 OK
```
```
{}  
```
Admin - Rename

Required permission: admin.tag

PUT-Request sent: /api/v1/tag_list/{tag id}

```json
{
    "name": "order"
}
```

Response:

```
HTTP/1.1 200 OK
{}
```

Admin - Delete

Required permission: admin.tag

DELETE-Request sent: /api/v1/tag_list/{tag id}

Response:

```
HTTP/1.1 200 OK
{}
```

23.6 Linking Tickets

23.6.1 Get

Required permission: ticket.agent or admin

GET-Request sent: /api/v1/links

```json
{
    "link_object": "Ticket",
    "link_object_value": "147469"
}
```

Response:

```
HTTP/1.1 200 OK
{
    "links": [
        {
            "link_type": "normal",
            "link_object": "Ticket",
            "link_object_value": 147470
        }
    ]
}
```
{  
"link_type": "normal",
"link_object": "Ticket",
"link_object_value": 147471
}
],
"assets": {
"Ticket": {
"147470": {
"id": 147470,
"group_id": 1,
"priority_id": 2,
"state_id": 2,
"organization_id": null,
"number": "34147470",
"title": "Test Ticket #2",
"owner_id": 1,
"customer_id": 3,
"note": null,
"first_response_at": null,
"first_response_escalation_at": null,
"first_response_in_min": null,
"first_response_diff_in_min": null,
"close_at": null,
"close_escalation_at": null,
"close_in_min": null,
"close_diff_in_min": null,
"update_escalation_at": null,
"update_in_min": null,
"update_diff_in_min": null,
"last_contact_at": "2021-11-03T10:16:45.266Z",
"last_contact_agent_at": null,
"last_contact_customer_at": "2021-11-03T10:10:45.266Z",
"last_owner_update_at": null,
"create_article_type_id": 5,
"create_article_sender_id": 2,
"article_count": 1,
"pending_time": null,
"type": null,
"time_unit": null,
"preferences": {},
"updated_by_id": 3,
"created_by_id": 3,
"created_at": "2021-11-03T10:16:45.092Z",
"updated_at": "2021-11-03T10:17:01.428Z",
"remote_access_permission_by": null,
"remote_access": "",
"affected_area": "",
"service_number": "",
"article_ids": [464964]"}}}
"ticket_time_accounting_ids": []
},
"147471": {
"id": 147471,
"group_id": 1,
"priority_id": 2,
"state_id": 2,
"organization_id": null,
"number": "34147471",
"title": "Test Ticket #3",
"owner_id": 1,
"customer_id": 3,
"note": null,
"first_response_at": null,
"first_response_escalation_at": null,
"first_response_in_min": null,
"first_response_diff_in_min": null,
"close_at": null,
"close_escalation_at": null,
"close_in_min": null,
"close_diff_in_min": null,
"update_escalation_at": null,
"update_in_min": null,
"update_diff_in_min": null,
"last_contact_at": "2021-11-03T10:16:51.995Z",
"last_contact_agent_at": null,
"last_contact_customer_at": "2021-11-03T10:16:51.995Z",
"last_owner_update_at": null,
"create_article_type_id": 5,
"create_article_sender_id": 2,
"article_count": 1,
"escalation_at": null,
"pending_time": null,
"type": null,
"time_unit": null,
"preferences": {},
"updated_by_id": 3,
"created_by_id": 3,
"created_at": "2021-11-03T10:16:51.824Z",
"updated_at": "2021-11-03T10:16:57.862Z",
"remote_access_permission_by": null,
"remote_access": 
"affected_area": 
"service_number": 
"article_ids": [
  464965
],
"ticket_time_accounting_ids": []
}
},
"User": {

"3": {
  "id": 3,
  "organization_id": null,
  "login": "test@test.de",
  "firstname": "Max",
  "lastname": "Mustermann",
  "email": "test@test.de",
  "image": null,
  "image_source": null,
  "web": "",
  "phone": "",
  "fax": "",
  "mobile": "",
  "department": null,
  "street": "",
  "zip": "",
  "city": "",
  "country": "",
  "address": null,
  "vip": false,
  "verified": false,
  "active": true,
  "note": "",
  "last_login": "2021-11-01T05:03:23.328Z",
  "source": null,
  "login_failed": 0,
  "out_of_office": false,
  "out_of_office_start_at": null,
  "out_of_office_end_at": null,
  "out_of_office_replacement_id": null,
  "preferences": {
    "notification_config": {
      "matrix": {
        "criteria": {
          "owned_by_me": true,
          "owned_by_nobody": true,
          "subscribed": true,
          "no": false
        },
        "channel": {
          "email": true,
          "online": true
        }
      }},
    "update": {
      "criteria": {
        "owned_by_me": true,
        "owned_by_nobody": true,
        "subscribed": true,
        "no": false
      }
    }
  }
}
"channel": {  
  "email": true,  
  "online": true
},
"reminder_reached": {  
  "criteria": {  
    "owned_by_me": true,  
    "owned_by_nobody": false,  
    "subscribed": false,  
    "no": false
  },  
  "channel": {  
    "email": true,  
    "online": true
}
},
"escalation": {  
  "criteria": {  
    "owned_by_me": true,  
    "owned_by_nobody": false,  
    "subscribed": false,  
    "no": false
  },  
  "channel": {  
    "email": true,  
    "online": true
}
},
"locale": "de-de",
"intro": true
},
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-10-29T13:25:59.261Z",
"updated_at": "2021-11-01T05:03:23.334Z",
"salutation": null,
"wawi_number": 0,
"guid": null,
"notification_optin": false,
"zr_number": null,
"role_ids": [1, 2],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": { "1": [  
  
  ] }
"full"
]
},
"accounts": {},
},
"1": {
"id": 1,
"organization_id": null,
"login": "-",
"firstname": "-",
"lastname": "",
"email": ""
"image": null,
"image_source": null,
"web": "",
"phone": "",
"fax": "",
"mobile": "",
"department": "",
"street": "",
"zip": "",
"city": "",
"country": "",
"address": "",
"vip": false,
"verified": false,
"active": false,
"note": "",
"last_login": null,
"source": null,
"login_failed": 0,
"out_of_office": false,
"out_of_office_start_at": null,
"out_of_office_end_at": null,
"out_of_office_replacement_id": null,
"preferences": {},
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-10-29T13:11:53.378Z",
"updated_at": "2021-10-29T13:11:53.378Z",
"salutation": null,
"wawi_number": 0,
"guid": null,
"notification_optin": false,
"zr_number": null,
"role_ids": [],
"organization_ids": [],
"authorization_ids": [],
"karma_user_ids": [],
"group_ids": {},
"accounts": {}
}
},
  "Role": {
    "1": {
      "id": 1,
      "name": "Admin",
      "preferences": {},
      "default_at_signup": false,
      "active": true,
      "note": "To configure your system.",
      "updated_by_id": 3,
      "created_by_id": 1,
      "created_at": "2021-10-29T13:11:53.503Z",
      "updated_at": "2021-10-30T21:44:00.923Z",
      "permission_ids": [
        1,
        41,
        51,
        61
      ],
      "group_ids": {}
    },
    "2": {
      "id": 2,
      "name": "Agent",
      "preferences": {},
      "default_at_signup": false,
      "active": true,
      "note": "To work on Tickets.",
      "updated_by_id": 3,
      "created_by_id": 1,
      "created_at": "2021-10-29T13:11:53.542Z",
      "updated_at": "2021-10-30T21:44:00.923Z",
      "permission_ids": [
        41,
        53,
        56,
        58,
        62
      ],
      "group_ids": {}
    }
  },
  "Group": {
    "1": {
      "id": 1,
      "signature_id": 1,
      "email_address_id": null,
      "name": "Users",
      "assignment_timeout": null,
      "follow_up_possible": "yes",
      "follow_up_assignment": true,
      "active": true,
      "note": "To configure your system."
    }
  }
}
"note": "Standard Group/Pool for Tickets.",
"updated_by_id": 1,
"created_by_id": 1,
"created_at": "2021-10-29T13:11:54.863Z",
"updated_at": "2021-10-30T18:53:24.803Z",
"user_ids": [
    3,
    2,
    5,
    55,
    65,
    83,
    90,
    101,
    105,
    112,
    118,
    132,
    153,
    168,
    203,
    295,
    493,
    515,
    528,
    535,
    565,
    618,
    730,
    755,
    839,
    859,
    912,
    914,
    983,
    1106,
    1138,
    1229,
    1287,
    1405,
    1410,
    1482,
    1486,
    1490,
    1543,
    1573
]
### 23.6.2 Add

Required permission: `ticket.agent` or `admin`

POST-Request sent: `/api/v1/links/add`

```
{
    "link_type": "normal",
    "link_object_target": "Ticket",
    "link_object_target_value": "147469",
    "link_object_source": "Ticket",
    "link_object_source_number": "34147471"
}
```

Response:

```
HTTP/1.1 201 Created

{
    "id": 5,
    "link_type_id": 1,
    "link_object_source_id": 1,
    "link_object_source_value": 147471,
    "link_object_target_id": 1,
    "link_object_target_value": 147469,
    "created_at": "2021-11-03T10:40:14.528Z",
    "updated_at": "2021-11-03T10:40:14.528Z"
}
```

### 23.6.3 Delete

Required permission: `ticket.agent` or `admin`

DELETE-Request sent: `/api/v1/links/remove`

```
{
    "link_type": "normal",
    "link_object_source": "Ticket",
    "link_object_source_value": 147471,
    "link_object_target": "Ticket",
    "link_object_target_value": 147469
}
```

Response:

```
HTTP/1.1 201 Created

{
}
```
23.7 Mentions

**Warning:** Mention endpoints depend on group permissions if the user you’re using is an agent. Because of this tickets may or may not be available.

### 23.7.1 List

Required permission: `ticket.agent` or `ticket.customer`

GET-Request sent: `/api/v1/mentions`

```json
HTTP/1.1 200 OK

{ 
  "mentions": [
  
    { 
      "id":2,
      "mentionable_type":"Ticket",
      "mentionable_id":1,
      "user_id":3,
      "updated_by_id":3,
      "created_by_id":3,
      "created_at":"2021-03-16T08:51:08.985Z",
      "updated_at":"2021-03-16T08:51:08.985Z"
    },
    
    { 
      "id":3,
      "mentionable_type":"Ticket",
      "mentionable_id":1,
      "user_id":4,
      "updated_by_id":4,
      "created_by_id":4,
      "created_at":"2021-03-16T08:51:08.986Z",
      "updated_at":"2021-03-16T08:51:08.986Z"
    }
  
  }
}
```

### 23.7.2 Create

Required permission: `ticket.agent`

POST-Request sent: `/api/v1/mentions`

```json
{
  "mentionable_type": "Ticket",
  "mentionable_id": 12,
}
```

Response:
The mention will be created for the user of the current session.

### 23.7.3 Delete

Required permission: `ticket.agent`

DELETE-Request sent: `/api/v1/mentions/{id}`

Response:
```json
HTTP/1.1 200 OK

{
    "id":2,
    "mentionable_type":"Ticket",
    "mentionable_id":1,
    "user_id":3,
    "updated_by_id":3,
    "created_by_id":3,
    "created_at":"2021-03-16T08:51:08.985Z",
    "updated_at":"2021-03-16T08:51:08.985Z"
}
```
This page describes the generic CTI API scopes and functionalities.

Features  Here’s a small condensed list of the possibilities this CTI API provides.

Inbound
- Caller log functions for your agents.
- Blocking of CallerIDs during signaling. *

Outbound
- Caller log functions for your agents.
- Set outbound caller IDs depending on the caller ID target.

Warning: Limitations / Notes
- Authentication on this endpoint works fundamentally different compared to the rest of the API.
- API clients do not work with the CTI endpoints unless explicitly stated by the client vendor!
- The CTI endpoints are relevant for PBX systems only.

Endpoint  The endpoint can be found in the generic CTI integration and contains a unique token which acts as authentication. Make sure to keep this endpoint URL safe.

Hint:  Generic CTI configuration and the correct endpoint can be found in your Zammad integration settings and are documented in our admin documentation.

Please also note the there listed requirements and limitations.
All options that require returns (e.g. blocking, manipulating outgoing caller IDs) rely on configurations within the Zammad CTI integration page.

Events  There are several events in terms of an ongoing call. These actions always come from your PBX system and may be:
- “newCall” event (initiation of a call)
- “hangup” event (call ending)
- “answer” event (aka picking up the phone)
In some situations Zammad may provide a return on your PBX calls (e.g. a reject) if you blocked a specific caller. Zammad will never initiate specific actions with your PBX. Zammad is a passive component in all described cases.

### 24.1 New call

**Warning: Limitations / Notes**

- Authentication on this endpoint works fundamentally different compared to the rest of the API.
- API clients *do not* work with the CTI endpoints unless explicitly stated by the client vendor!
- The CTI endpoints are relevant for PBX systems only.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Possible value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>event</td>
<td>newCall</td>
<td>Tell Zammad there’s a new call</td>
</tr>
<tr>
<td>from</td>
<td>e.g. 493055571600, 02214710334, anonymous</td>
<td>Number that initiated the call</td>
</tr>
<tr>
<td>to</td>
<td>e.g. 49221470334, 03023125771</td>
<td>Number that is being called</td>
</tr>
<tr>
<td>direction</td>
<td>in or out</td>
<td>The call direction - if your agent initiates a call this will be out</td>
</tr>
<tr>
<td>callId</td>
<td>e.g. 53ba82e2bd6d12d9fb2d3838f0c9a0f607b0f2b9532f40da8334a, 123456789</td>
<td>An ID that is unique for the call. Zammad will use this ID to identify an existing call with following actions (e.g. like answering or hanging up) <em>This ID must be unique per call session.</em></td>
</tr>
<tr>
<td>user</td>
<td>e.g. John Doe, [Alice, Bob]</td>
<td>The user(s) real name involved. You may have to provide array style ([ ]) params depending on the call method you choose. If the direction is out, this is the name of the calling person(s). If the direction is in, this is the name of the called person(s). This value is optional.</td>
</tr>
<tr>
<td>queue</td>
<td>e.g. support, sales</td>
<td>An optional queue name, this option is relevant for the <a href="#">Caller Log Filter</a></td>
</tr>
</tbody>
</table>

There’s two options on how to POST the relevant data to Zammad.

**Note: For your context**

Below calls have been sent with the following configuration. This is important for you to understand the returns we’re showing here.

**Outbound**

- Destination caller ID 4989* set outbound caller ID 498999998145 with note “All from munich”
- Destination caller ID 4930* set outbound caller ID 493023125877 “All from Berlin”
Other settings

- Default caller ID for outbound calls 496990009111

JSON (recommended)

form-data

POST-Request sent: https://{FQDN-Zammad}/api/v1/cti/{instance specific token}

**Outbound** Payload:

```javascript
{
   "event": "newCall",
   "from": "493023125741",
   "to": "492214710334",
   "direction": "out",
   "callId": "f4ebd2be-7b9a-4d58-94c2-eb06a3c2ce76",
   "user": "Christopher Miller"
}
```

Returns:

```javascript
{
   "action": "dial",
   "caller_id": "496990009111",
   "number": "492214710334"
}
```

Sample curl command:

```
$ curl --request POST 'https://{FQDN-Zammad}/api/v1/cti/{instance specific token}'
   --header 'Content-Type: application/json'
   --data-raw '{
   "event": "newCall",
   "from": "493023125741",
   "to": "492214710334",
   "direction": "out",
   "callId": "f4ebd2be-7b9a-4d58-94c2-eb06a3c2ce76",
   "user": "Christopher Miller"
}'
```

**Inbound** Payload:

```javascript
{
   "event": "newCall",
   "from": "493023125741",
   "to": "492214710334",
   "direction": "in",
   "callId": "307fa962-de8d-4ffc-817b-7f6993204159",
   "user": ["Christopher Miller", "Emma Taylor"]
}
```

Response:
Sample curl command:

```
$ curl --request POST 'https://{FQDN-Zammad}/api/v1/cti/{instance specific → token}' \
    --header 'Content-Type: application/json' \
    --data-raw '{
        "event": "newCall",
        "from": "493023125741",
        "to": "492214710334",
        "direction": "in",
        "callId": "307fa962-de8d-4f6f-817b-7f6993204159",
        "user": ["Christopher Miller", "Emma Taylor"]
    }'
```

Outbound Payload:

```
event:"newCall" from:"493023125741" to:"492214710334" direction:"out" callId:"f0871278-0600-4f5c-a746-bec3acf04f41" user:"Christopher Miller"
```

Returns:

```
{
    "action": "dial",
    "caller_id": "496990009111",
    "number": "492214710334"
}
```

Sample curl command:

```
$ curl --request POST 'https://{FQDN-Zammad}/api/v1/cti/{instance specific → token}' \
    --form 'event=newCall' \
    --form 'from=493023125741' \
    --form 'to=492214710334' \
    --form 'direction=out' \
    --form 'callId=f0871278-0600-4f5c-a746-bec3acf04f41' \
    --form 'user=Christopher Miller'
```

Inbound Payload:

```
event:"newCall" from:"493023125741" to:"492214710334" direction:"in" callId:"25641e3f-3317-4c48-80b3-fc573c7ffe2b" user[]:"Christopher Miller" user[]:"Emma Taylor"
```
Returns:

```{}
```

Sample curl command:

```bash
$ curl --request POST 'https://{FQDN-Zammad}/api/v1/cti/{instance specific token}' \
--form 'event="newCall"' \
--form 'from="493023125741"' \
--form 'to="492214710334"' \
--form 'direction="in"' \
--form 'callId="25641e3f-3317-4c48-80b3-fc573c7ffe2b"' \
--form 'user[]="Christopher Miller"' \
--form 'user[]="Emma Taylor"'
```

### 24.1.1 Situation specific responses

Depending on the chosen call direction, Zammad will return either a (optionally) configured call ID or (optionally) block a caller. If your Zammad hasn’t configured one or both options, the return will be empty.

**Note:** This has to be supported by your PBX in order to work.

Reject blocked caller IDs

Set specific outgoing caller ID

If an incoming new call matches a to block number, Zammad will return the following.

```json
{
   "action": "reject",
   "reason": "busy"
}
```

If no to block number matches, Zammad will return the following.

```{}
```

**Warning:** Your PBX still needs to end the call (hangup event). Other wise the call will not just appear within Zammad’s caller log but also appear as ringing call.

In case your instance has a matching overwriting caller ID configured, Zammad will return the following payload.

```json
{
   "action": "dial",
   "callerId": "493055571642",
   "number": "491711234567890"
}
```

If no overwrite match is found or you haven’t configured anything, Zammad will return the following.
The next logical steps within call session context would be:

- `hangup` (call ending)
- `answer` (aka picking up the phone)

### 24.2 Call hangup

<table>
<thead>
<tr>
<th>Warning: Limitations / Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Authentication on this endpoint works fundamentally different compared to the rest of the API.</td>
</tr>
<tr>
<td>• API clients <em>do not</em> work with the CTI endpoints unless explicitly stated by the client vendor!</td>
</tr>
<tr>
<td>• The CTI endpoints are relevant for PBX systems only.</td>
</tr>
</tbody>
</table>
Table 2: Available attributes and sample data for hangup events

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Possible value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>event</td>
<td>hangup</td>
<td>Tell Zammad that somebody hung up the call.</td>
</tr>
<tr>
<td>from</td>
<td>e.g. 493055571600, 02214710334, anonymous</td>
<td>Number that initiated the call</td>
</tr>
<tr>
<td>to</td>
<td>e.g. 49221470334, 03023125771</td>
<td>Number that is being called</td>
</tr>
<tr>
<td>direction</td>
<td>in or out</td>
<td>The call direction - if your agent initiates a call this will be out</td>
</tr>
<tr>
<td>callId</td>
<td>e.g. 53ba82e2bd6d12d9fb2d3838f0cfb0f75fb9532f40da834a, 123456789</td>
<td>An ID that is unique for the call. Zammad will use this ID to identify an existing call with following actions (e.g. like answering or hanging up). <em>This ID must be unique per call session.</em></td>
</tr>
<tr>
<td>cause</td>
<td></td>
<td>This defines the reason of the hangup. Zammad evaluates the cause and indicates e.g. missed calls accordingly in the caller log.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Cause type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>normalClearing</td>
<td>The parties hung up after the call was established.</td>
</tr>
<tr>
<td>busy</td>
<td>The called party was busy</td>
</tr>
<tr>
<td>cancel</td>
<td>The caller hung up before the called party picked up</td>
</tr>
<tr>
<td>noAnswer</td>
<td>The called party rejected the call (e.g. through a DND setting)</td>
</tr>
<tr>
<td>congestion</td>
<td>The called party could not be reached</td>
</tr>
<tr>
<td>notFound</td>
<td>The called number does not exist or called party is offline</td>
</tr>
<tr>
<td>forwarded</td>
<td>The call was forwarded to a different party</td>
</tr>
</tbody>
</table>

| answeringNumber | e.g. 42, jdoe, jdoe@example.com, 3 | Zammad will look up for a user with given value, the following attributes will be evaluated in given order:  
- user.phone  
- user.login  
- user.id  
This value is optional. |

There’s two options on how to POST the relevant data to Zammad.

**Note:** For your context

Below calls have been sent with the following configuration. This is important for you to understand the returns we’re showing here.

**Outbound**

- Destination caller ID 4989* set outbound caller ID 498999998145 with note “All from munich”
- Destination caller ID 4930* set outbound caller ID 493023125877 “All from Berlin”

**Other settings**
Zammad

- Default caller ID for outbound calls 496990009111

JSON (recommended)

form-data

POST-Request send: https://{FQDN-Zammad}/api/v1/cti/{instance specific token}

**Outbound** Payload:

```json
{
    "event": "hangup",
    "from": "493023125741",
    "to": "492214710334",
    "direction": "out",
    "callId": "f4ebd2be-7b9a-4d58-94c2-eb06a3c2ce76",
    "cause": "cancel"
}
```

Response:

```
{}
```

Sample curl command:

```
$ curl --request POST 'https://{FQDN-Zammad}/api/v1/cti/{instance specific token}' \
   --header 'Content-Type: application/json' \
   --data-raw '{
       "event": "hangup",
       "from": "493023125741",
       "to": "492214710334",
       "direction": "out",
       "callId": "f4ebd2be-7b9a-4d58-94c2-eb06a3c2ce76",
       "cause": "cancel"
   }
```

**Inbound** Payload:

```json
{
    "event": "hangup",
    "from": "493023125741",
    "to": "492214710334",
    "direction": "in",
    "callId": "307fa962-de8d-4ffc-817b-7f6993204159",
    "answeringNumber": "emma@chrispresso.com",
    "cause": "normalClearing"
}
```

Response:

```
{}
```

Sample curl command:
Zammad

$ curl --request POST 'https://{FQDN-Zammad}/api/v1/cti/{instance specific token}' \
   --header 'Content-Type: application/json' \
   --data-raw '{
      "event": "hangup",
      "from": "493023125741",
      "to": "492214710334",
      "direction": "in",
      "callId": "307fa962-de8d-4fff-817b-7f6993204159",
      "answeringNumber": "emma@chrispresso.com",
      "cause": "normalClearing"
   }'

POST-Request sent: https://{FQDN-Zammad}/api/v1/cti/{instance specific token}

**Outbound** Payload:

```
event:"hangup"
from:"493023125741"
to:"492214710334"
direction:"out"
callId:"da7cf8b8-2de2-4120-93c8-7db1f55225dc"
cause:"cancel"
```

Returns:

```
{}
```

Sample curl command:

```
$ curl --request POST 'https://{FQDN-Zammad}/api/v1/cti/{instance specific token}' \
   --form 'event="hangup"' \
   --form 'from="493023125741"' \
   --form 'to="492214710334"' \
   --form 'direction="out"' \
   --form 'callId="da7cf8b8-2de2-4120-93c8-7db1f55225dc"' \
   --form 'cause="cancel"'
```

**Inbound** Payload:

```
event:"hangup"
from:"493023125741"
to:"492214710334"
direction:"in"
callId:"2d77882f-68df-40f0-8c62-b642589c00bc"
answeringNumber:"emma@chrispresso.com",
cause:"normalClearing"
```

Returns:

```
{}
```

Sample curl command:

```
$ curl --request POST "https://{FQDN-Zammad}/api/v1/cti/{instance specific token}" \
        --form 'event="hangup"' \
        --form 'from="493023125741"' \
        --form 'to="492214710334"' \
        --form 'direction="in"' \
        --form 'callId="2d77882f-68df-40f0-8c62-b642589c00bc"' \
        --form 'answeringNumber="emma@chrispresso.com"' \
        --form 'cause="normalClearing"'

### 24.3 Call answered

<table>
<thead>
<tr>
<th>Warning: Limitations / Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Authentication on this endpoint works fundamentally different compared to the rest of the API.</td>
</tr>
<tr>
<td>• API clients <em>do not</em> work with the CTI endpoints unless explicitly stated by the client vendor!</td>
</tr>
<tr>
<td>• The CTI endpoints are relevant for PBX systems only.</td>
</tr>
</tbody>
</table>
### Table 3: Available attributes and sample data for answered events

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Possible value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>event</td>
<td>answer</td>
<td>Tell Zammad that someone answered the call.</td>
</tr>
<tr>
<td>from</td>
<td>e.g. 493055571600, 02214710334, anonymous</td>
<td>Number that initiated the call</td>
</tr>
<tr>
<td>to</td>
<td>e.g. 49221470334, 03023125771</td>
<td>Number that is being called</td>
</tr>
<tr>
<td>direction</td>
<td>in or out</td>
<td>The call direction - if your agent initiates a call this will be out</td>
</tr>
</tbody>
</table>
| callId       | e.g. 53ba82e2bd6d12d9fb2d3838f0cfb0705fb9532f40da834a, 123456789 | An ID that is unique for the call. Zammad will use this ID to identify an existing call with following actions (e.g. like answering or hanging up)
|              |                | This ID must be unique per call session.                                    |
| answeringNumber | e.g. 42, jdoe, jdoe@example.com, 49221470351, 03023125184 | Zammad will look up for a user with given value, the following attributes will be evaluated in given order:  
|              |                | user.phone, user.login, user.id  
|              |                | This value is optional.                                                     |
| user         | e.g. John Doe, [Alice, Bob] | The user(s) real name involved. You may have to provide array style ([]) params depending on the call method you choose.  
|              |                | If the direction is out, this is the name of the calling person(s).         |
|              |                | If the direction is in, this is the name of the called person(s).           |
|              |                | This value is optional.                                                     |

There’s two options on how to POST the relevant data to Zammad.

**Note:** For your context

Below calls have been sent with the following configuration. This is important for you to understand the returns we’re showing here.

**Outbound**

- Destination caller ID 4989* set outbound caller ID 498999998145 with note “All from munich”
- Destination caller ID 4930* set outbound caller ID 493023125877 “All from Berlin”

**Other settings**

- Default caller ID for outbound calls 496990009111

JSON (recommended)

form-data

POST-Request sent: https://{FQDN-Zammad}/api/v1/cti/{instance specific token}

Outbound Payload:
Zammad

```json
{
    "event": "answer",
    "from": "493023125741",
    "to": "492214710334",
    "direction": "out",
    "callId": "9f1840cb-8be9-4d3a-8200-3da2937085f0",
    "caller": "Christopher Miller"
}
```

Response:

```json
{}
```

Sample curl command:

```bash
$ curl --request POST 'https://{FQDN-Zammad}/api/v1/cti/{instance specific token}' \
   --header 'Content-Type: application/json' \
   --data-raw '{
       "event": "answer",
       "from": "493023125741",
       "to": "492214710334",
       "direction": "out",
       "callId": "9f1840cb-8be9-4d3a-8200-3da2937085f0",
       "caller": "Christopher Miller"
   }'
```

Inbound Payload:

```json
{
    "event": "answer",
    "from": "493023125741",
    "to": "492214710334",
    "direction": "in",
    "callId": "307fa962-de8d-4ffc-817b-7f6993204159",
    "answeringNumber": "emma@chrispresso.com",
    "caller": ["Christopher Miller", "Emma Taylor"]
}
```

Response:

```json
{}
```

Sample curl command:

```bash
$ curl --request POST 'https://{FQDN-Zammad}/api/v1/cti/{instance specific token}' \
   --header 'Content-Type: application/json' \
   --data-raw '{
       "event": "answer",
       "from": "493023125741",
       "to": "492214710334",
       "direction": "in",
       "callId": "307fa962-de8d-4ffc-817b-7f6993204159",
   }'
```

(continues on next page)
"answeringNumber": "emma@chrispresso.com",
"caller": ["Christopher Miller", "Emma Taylor"]
}

POST-Request sent: https://{FQDN-Zammad}/api/v1/cti/{instance specific token}

**Outbound** Payload:

```json
event:"answer"
from:"493023125741"
to:"492214710334"
direction:"out"
callId:"371e2cd7-67ff-4fd9-892b-030c8d128fb1"
caller[]:"Christopher Miller"
caller[]:"Emma Taylor"
```

Returns:

```json
{}
```

Sample curl command:

```
$ curl --request POST 'https://{FQDN-Zammad}/api/v1/cti/{instance specific token}'
  --form 'event="answer"'
  --form 'from="493023125741"'
  --form 'to="492214710334"'
  --form 'direction="out"'
  --form 'callId="371e2cd7-67ff-4fd9-892b-030c8d128fb1"'
  --form 'caller[]="Christopher Miller"'
  --form 'caller[]="Emma Taylor"'
```

**Inbound** Payload:

```json
event:"answer"
from:"493023125741"
to:"492214710334"
direction:"in"
callId:"61868f1e-2171-4313-970b-25982f0c5ce1"
answeringNumber:"emma@chrispresso.com"
caller:"Emma Taylor"
```

Returns:

```json
{}
```

Sample curl command:

```
$ curl --request POST 'https://{FQDN-Zammad}/api/v1/cti/{instance specific token}'
  --form 'event="answer"'
  --form 'from="493023125741"'
  --form 'to="492214710334"'
  --form 'direction="in"'
```

(continues on next page)
The next logical step within call session context would be:

- **hangup** (call ending)
Note: The availability of notification highly depends on the users permission and chosen notification settings. Please note that the best results are always achieved with Agents.

25.1 List

Required permission: any

Tip: Use the expand request to know the affected objects. Otherwise you’ll need to find out what ID stands for which object type.

GET-Request sent: /api/v1/online_notifications?expand=true

Response:

```
HTTP/1.1 200 OK

[
  {
    "id": 4,
    "o_id": 6,
    "object_lookup_id": 2,
    "type_lookup_id": 1,
    "user_id": 3,
    "seen": false,
    "updated_by_id": 8,
    "created_by_id": 8,
    "created_at": "2021-11-09T13:15:42.628Z",
    "updated_at": "2021-11-09T13:15:42.637Z",
    "user": "chris@chrispresso.com",
    "object": "Ticket",
    "type": "create",
    "created_by": "emily@example.com",
    "updated_by": "emily@example.com"
  },
  {
    "id": 3,
    "o_id": 6,
    "object_lookup_id": 2,
    "type_lookup_id": 1,
    "user_id": 3,
    "seen": true,
    "updated_by_id": 8,
    "created_by_id": 8,
    "created_at": "2021-11-09T13:15:42.628Z",
    "updated_at": "2021-11-09T13:15:42.637Z",
    "user": "chris@chrispresso.com",
    "object": "Ticket",
    "type": "create",
    "created_by": "emily@example.com",
    "updated_by": "emily@example.com"
  }
]
```
"o_id": 8,
"object_lookup_id": 2,
"type_lookup_id": 2,
"user_id": 3,
"seen": false,
"updated_by_id": 4,
"created_by_id": 4,
"created_at": "2021-11-09T13:10:42.628Z",
"updated_at": "2021-11-09T13:15:42.635Z",
"user": "chris@chrispresso.com",
"object": "Ticket",
"type": "update",
"created_by": "jacob@chrispresso.com",
"updated_by": "jacob@chrispresso.com"
},
{
"id": 2,
"o_id": 3,
"object_lookup_id": 2,
"type_lookup_id": 1,
"user_id": 3,
"seen": true,
"updated_by_id": 6,
"created_by_id": 6,
"created_at": "2021-11-09T12:45:42.625Z",
"updated_at": "2021-11-09T13:15:42.632Z",
"user": "chris@chrispresso.com",
"object": "Ticket",
"type": "create",
"created_by": "anna@example.com",
"updated_by": "anna@example.com"
},
{
"id": 1,
"o_id": 2,
"object_lookup_id": 2,
"type_lookup_id": 1,
"user_id": 3,
"seen": true,
"updated_by_id": 5,
"created_by_id": 5,
"created_at": "2021-11-09T11:45:42.624Z",
"updated_at": "2021-11-09T13:15:42.629Z",
"user": "chris@chrispresso.com",
"object": "Ticket",
"type": "create",
"created_by": "emma@chrispresso.com",
"updated_by": "emma@chrispresso.com"
}
25.2 Show

Required permission: any

GET-Request sent: /api/v1/online_notifications/{id}

Response:

```
# HTTP Code 200 OK
{
  "id": 4,
  "o_id": 6,
  "object_lookup_id": 2,
  "type_lookup_id": 1,
  "user_id": 3,
  "seen": false,
  "updated_by_id": 8,
  "created_by_id": 8,
  "created_at": "2021-11-09T13:15:42.628Z",
  "updated_at": "2021-11-09T13:15:42.637Z"
}
```

25.3 Update

Required permission: any

PUT-Request sent: /api/v1/online_notifications/{id}

```
{
  "seen": true
}
```

Response:

```
# HTTP Code 200 OK
{
  "id": 4,
  "seen": true,
  "updated_by_id": 3,
  "o_id": 6,
  "object_lookup_id": 2,
  "type_lookup_id": 1,
  "user_id": 3,
  "created_by_id": 8,
  "created_at": "2021-11-09T13:15:42.628Z",
  "updated_at": "2021-11-09T13:25:00.004Z"
}
```
25.4 Delete

Required permission: any

DELETE-Request sent: /api/v1/online_notifications/{id}

Response:

```
HTTP CODE 200 OK
{}
```

25.5 Mark all as read

Required permission: any

POST-Request sent: /api/v1/online_notifications/mark_all_as_read

Response:

```
HTTP CODE 200 OK
{}
```
**Danger:** Adjusting objects via API can cause serious issues with your instance. Proceed with absolute caution and ensure to adjust any of Zammads default fields.

If you want to hide fields, consider Core Workflows instead. For states and priorities use either API endpoints or rails console.

### 26.1 List

Required permission: admin.object

GET-Request sent: /api/v1/object_manager_attributes

Response:

```json
# HTTP Code 200 OK
[
  {
    "id": 2,
    "name": "customer_id",
    "display": "Customer",
    "data_type": "user_autocompletion",
    "data_option": {
      "relation": "User",
      "autocapitalize": false,
      "multiple": false,
      "guess": true,
      "null": false,
      "limit": 200,
      "placeholder": "Enter Person or Organization/Company",
      "minLengt": 2,
      "translate": false,
      "permission": [
        "ticket.agent"
      ]
    },
    "data_option_new": {},
    "editable": false,
    "active": true,
  }
] (continues on next page)```
"screens": {
  "create_top": {
    "-all-": {
      "null": false
    },
  },
  "edit": {}
},
"to_create": false,
"to_migrate": false,
"to_delete": false,
"to_config": false,
"position": 10,
"created_by_id": 1,
"updated_by_id": 1,
"created_at": "2021-11-09T13:12:32.677Z",
"updated_at": "2021-11-09T13:12:32.677Z",
"object": "Ticket",
"deletable": false,
"not_deletable_reason": "This attribute is referenced by Overview: My Tickets and thus cannot be deleted!"
},
{
  "id": 1,
  "name": "title",
  "display": "Title",
  "data_type": "input",
  "data_option": {
    "type": "text",
    "maxlength": 200,
    "null": false,
    "translate": false
  },
  "data_option_new": {},
  "editable": false,
  "active": true,
  "screens": {
    "create_top": {
      "-all-": {
        "null": false
      },
    },
    "edit": {}
  },
  "to_create": false,
  "to_migrate": false,
  "to_delete": false,
  "to_config": false,
  "position": 15,
  "created_by_id": 1,
  "updated_by_id": 1,
  "created_at": "2021-11-09T13:12:32.671Z"
"updated_at": "2021-11-09T13:12:32.671Z",
"object": "Ticket",
"deletable": false
},
{
  "id": 3,
  "name": "type",
  "display": "Type",
  "data_type": "select",
  "data_option": {
    "default": ",",
    "options": {
      "Incident": "Incident",
      "Problem": "Problem",
      "Request for Change": "Request for Change"
    },
    "nulloption": true,
    "multiple": false,
    "null": true,
    "translate": true,
    "maxlength": 255
  },
  "data_option_new": {},
  "editable": true,
  "active": false,
  "screens": {
    "create_middle": {
      "-all-": {
        "null": false,
        "item_class": "column"
      }
    },
    "edit": {
      "ticket.agent": {
        "null": false
      }
    }
  },
  "to_create": false,
  "to_migrate": false,
  "to_delete": false,
  "to_config": false,
  "position": 20,
  "created_by_id": 1,
  "updated_by_id": 1,
  "created_at": "2021-11-09T13:12:32.686Z",
  "updated_at": "2021-11-09T13:12:32.686Z",
  "object": "Ticket",
  "deletable": true
},
{
  "id": 4,
"name": "group_id",
"display": "Group",
"data_type": "select",
"data_option": {
  "default": "",
  "relation": "Group",
  "relation_condition": {
    "access": "full"
  },
  "nulloption": true,
  "multiple": false,
  "null": false,
  "translate": false,
  "only_shown_if_selectable": true,
  "permission": [
    "ticket.agent",
    "ticket.customer"
  ],
  "maxlength": 255
},
"data_option_new": {},
"editable": false,
"active": true,
"screens": {
  "create_middle": {
    "-all-": {
      "null": false,
      "item_class": "column"
    },
    "edit": {
      "ticket.agent": {
        "null": false
      }
    }
  }
},
"to_create": false,
"to_migrate": false,
"to_delete": false,
"to_config": false,
"position": 25,
"created_by_id": 1,
"updated_by_id": 1,
"created_at": "2021-11-09T13:12:32.690Z",
"updated_at": "2021-11-09T13:12:32.690Z",
"object": "Ticket",
"deletable": false
},
{
  "id": 5,
  "name": "owner_id",
  "display": "Owner",
  "data_type": "select",
  "data_option": {
    "default": "",
    "relation": "Owner",
    "relation_condition": {
      "access": "full"
    },
    "nulloption": true,
    "multiple": false,
    "null": false,
    "translate": false,
    "only_shown_if_selectable": true,
    "permission": [
      "ticket.agent",
      "ticket.customer"
    ],
    "maxlength": 255
  },
  "data_option_new": {},
  "editable": false,
  "active": true,
  "screens": {
    "create_middle": {
      "-all-": {
        "null": false,
        "item_class": "column"
      },
      "edit": {
        "ticket.agent": {
          "null": false
        }
      }
    }
  },
  "to_create": false,
  "to_migrate": false,
  "to_delete": false,
  "to_config": false,
  "position": 25,
  "created_by_id": 1,
  "updated_by_id": 1,
  "created_at": "2021-11-09T13:12:32.690Z",
  "updated_at": "2021-11-09T13:12:32.690Z",
  "object": "Ticket",
  "deletable": false
}
"data_type": "select",
"data_option": {
  "default": "",
  "relation": "User",
  "relation_condition": {
    "roles": "Agent"
  },
  "nulloption": true,
  "multiple": false,
  "null": true,
  "translate": false,
  "permission": [
    "ticket.agent"
  ],
  "maxlength": 255
},
"data_option_new": {},
"editable": false,
"active": true,
"screens": {
  "create_middle": {
    "-all-": {
      "null": true,
      "item_class": "column"
    }
  },
  "edit": {
    "-all-": {
      "null": true
    }
  }
},
"to_create": false,
"to_migrate": false,
"to_delete": false,
"to_config": false,
"position": 30,
"created_by_id": 1,
"updated_by_id": 1,
"created_at": "2021-11-09T13:12:32.694Z",
"updated_at": "2021-11-09T13:12:32.694Z",
"object": "Ticket",
"deletable": false,
"not_deletable_reason": "This attribute is referenced by Trigger: customer␣
˓
notification (on owner change); Overview: My assigned Tickets, My pending reached␣
˓
Tickets, Unassigned & Open and thus cannot be deleted!"
},
{
  "id": 6,
  "name": "state_id",
  "display": "State",
  "data_type": "select",
  "data_option": {
    "default": "",
    "relation": "User",
    "relation_condition": {
      "roles": "Agent"
    },
    "nulloption": true,
    "multiple": false,
    "null": true,
    "translate": false,
    "permission": [
      "ticket.agent"
    ],
    "maxlength": 255
  }
}
"data_option": {
    "relation": "TicketState",
    "nulloption": true,
    "multiple": false,
    "null": false,
    "default": 2,
    "translate": true,
    "filter": [
        2,
        1,
        3,
        4,
        6,
        7
    ],
    "maxlength": 255
},
"data_option_new": {},
"editable": false,
"active": true,
"screens": {
    "create_middle": {
        "ticket.agent": {
            "null": false,
            "item_class": "column",
            "filter": [
                2,
                1,
                3,
                4,
                7
            ]
        },
        "ticket.customer": {
            "item_class": "column",
            "nulloption": false,
            "null": true,
            "filter": [
                1,
                4
            ],
            "default": 1
        }
    },
    "edit": {
        "ticket.agent": {
            "nulloption": false,
            "null": false,
            "filter": [
                2,
                3,
                4,
                5
            ]
        }
    }
}
Zammad

(continued from previous page)

7
]
},
"ticket.customer": {
"nulloption": false,
"null": true,
"filter": [
2,
4
],
"default": 2
}
}
},
"to_create": false,
"to_migrate": false,
"to_delete": false,
"to_config": false,
"position": 40,
"created_by_id": 1,
"updated_by_id": 1,
"created_at": "2021-11-09T13:12:32.706Z",
"updated_at": "2021-11-09T13:12:32.706Z",
"object": "Ticket",
"deletable": false,
"not_deletable_reason": "This attribute is referenced by Trigger: auto reply (on␣
˓→new tickets); Overview: My Organization Tickets,My Tickets,My assigned Tickets,My␣
˓→pending reached Tickets,My replacement Tickets,Open,Open Banana Items,Pending reached,
˓→Unassigned & Open,VIP Customers and thus cannot be deleted!"
},
{
"id": 7,
"name": "pending_time",
"display": "Pending till",
"data_type": "datetime",
"data_option": {
"future": true,
"past": false,
"diff": 24,
"null": true,
"translate": true,
"permission": [
"ticket.agent"
]
},
"data_option_new": {},
"editable": false,
"active": true,
"screens": {
"create_middle": {
"-all-": {
"null": false,
(continues on next page)

26.1. List

279


"item_class": "column"
}
},
"edit": {
  "-all-": {
    "null": false
  }
}
}
"to_create": false,
"to_migrate": false,
"to_delete": false,
"to_config": false,
"position": 41,
"created_by_id": 1,
"updated_by_id": 1,
"created_at": "2021-11-09T13:12:32.713Z",
"updated_at": "2021-11-09T13:12:32.713Z",
"object": "Ticket",
"deletable": false,
"not_deletable_reason": "This attribute is referenced by Overview: My pending reached Tickets,Pending reached and thus cannot be deleted!"
},
{
  "id": 8,
  "name": "priority_id",
  "display": "Priority",
  "data_type": "select",
  "data_option": {
    "relation": "TicketPriority",
    "nulloption": false,
    "multiple": false,
    "null": false,
    "default": 2,
    "translate": true,
    "maxlength": 255
  },
  "data_option_new": {},
  "editable": false,
  "active": true,
  "screens": {
    "create_middle": {
      "ticket.agent": {
        "null": false,
        "item_class": "column"
      }
    },
    "edit": {
      "ticket.agent": {
        "null": false
      }
    }
  }
}
26.2 Show

Required permission: admin.object

GET-Request sent: /api/v1/object_manager_attributes/{id}

Response:

```
HTTP/1.1 200 OK

{
  "id": 18,
  "object_lookup_id": 1,
  "name": "email",
  "display": "Email",
  "data_type": "input",
  "data_option": {
    "type": "email",
    "maxlength": 150,
    "null": true,
    "item_class": "formGroup--halfSize"
  },
  "data_option_new": {},
  "editable": false,
  "active": true,
  "screens": {
    "signup": {
      "-all-": {
        "null": false
      }
    },
    "invite_agent": {
      "-all-": {
        "null": false
      }
    }
  }
}
```
26.3 Create

Required permission: admin.object

POST-Request sent: /api/v1/object_manager_attributes

Boolean
Date
Date Time
Integer
Select
Text
Tree Select

Payload:
```json
{
  "name": "sample_boolean",
  "object": "Ticket",
  "display": "Sample Boolean",
  "active": true,
  "position": 1550,
  "data_type": "boolean",
  "data_option": {
    "options": {
      "true": "very correct indeed",
      "false": "very incorrect indeed"
    }
  },
  "screens": {
    "create_middle": {
      "ticket.customer": {
        "shown": true,
        "required": false,
        "item_class": "column"
      },
      "ticket.agent": {
        "shown": true,
        "required": false,
        "item_class": "column"
      }
    },
    "edit": {
      "ticket.customer": {
        "shown": true,
        "required": false
      },
      "ticket.agent": {
        "shown": true,
        "required": true
      }
    }
  }
}
```

Response:

```
# HTTP Code 201 Created

{
  "id": 50,
  "object_lookup_id": 2,
  "name": "sample_boolean",
  "display": "Sample Boolean",
  "data_type": "boolean",
  "data_option": {
    "options": {
      "true": "very correct indeed",
      "false": "very incorrect indeed"
    }
  }
}
```

(continues on next page)
Payload:

```json
{
  "name": "sample_date",
  "object": "Ticket",
  "display": "Sample Date",
  "active": true,
  "position": 1550,
  "created_by_id": 3,
  "updated_by_id": 3,
  "created_at": "2021-11-12T18:18:23.208Z",
  "updated_at": "2021-11-12T18:18:23.208Z"
}
```
"data_option": {
  "diff": 120
},
"screens": {
  "create_middle": {
    "ticket.customer": {
      "shown": true,
      "required": false,
      "item_class": "column"
    },
    "ticket.agent": {
      "shown": true,
      "required": false,
      "item_class": "column"
    }
  },
  "edit": {
    "ticket.customer": {
      "shown": true,
      "required": false
    },
    "ticket.agent": {
      "shown": true,
      "required": true
    }
  }
}

Response:

```
HTTP/1.1 201 Created

{
  "id": 51,
  "object_lookup_id": 2,
  "name": "sample_date",
  "display": "Sample Date",
  "data_type": "date",
  "data_option": {
    "diff": 120,
    "default": null,
    "null": true,
    "options": {},
    "relation": ""
  },
  "data_option_new": {},
  "editable": true,
  "active": true,
  "screens": {
    "create_middle": {
      "ticket.customer": {
        ...
```

(continues on next page)
"shown": true,
"required": false,
"item_class": "column"
},
"ticket.agent": {
  "shown": true,
  "required": false,
  "item_class": "column"
}
},
"edit": {
  "ticket.customer": {
    "shown": true,
    "required": false
  },
  "ticket.agent": {
    "shown": true,
    "required": true
  }
}
},
"to_create": true,
"to_migrate": true,
"to_delete": false,
"to_config": false,
"position": 1550,
"created_by_id": 3,
"updated_by_id": 3,
"created_at": "2021-11-12T18:19:32.827Z",
"updated_at": "2021-11-12T18:19:32.827Z"
}

Payload:

{
  "name": "sample_datetime",
  "object": "Ticket",
  "display": "Sample DateTime",
  "active": true,
  "position": 1550,
  "data_type": "datetime",
  "data_option": {
    "future": true,
    "past": false,
    "diff": 120
  },
  "screens": {
    "create_middle": {
      "ticket.customer": {
        "shown": true,
        "required": false,
        "item_class": "column"
      }
    }
  }
}
```
{
    "id": 52,
    "object_lookup_id": 2,
    "name": "sample_datetime",
    "display": "Sample DateTime",
    "data_type": "datetime",
    "data_option": {
        "future": true,
        "past": false,
        "null": true,
        "default": null,
        "options": {},
        "relation": ""
    },
    "data_option_new": {},
    "editable": true,
    "active": true,
    "screens": {
        "create_middle": {
            "ticket.customer": {
                "shown": true,
                "required": false,
                "item_class": "column"
            }
        },
        "ticket.agent": {
            "shown": true,
            "required": false,
            "item_class": "column"
        }
    }
}
```
Payload:

```json
{
    "name": "sample_integer",
    "object": "Ticket",
    "display": "Sample Integer",
    "active": true,
    "position": 1550,
    "data_type": "integer",
    "data_option": {
        "default": 1234,
        "min": 4,
        "max": 8
    },
    "screens": {
        "create_middle": {
            "ticket.customer": {
                "shown": true,
                "required": false,
                "item_class": "column"
            },
            "ticket.agent": {
                "shown": true,
                "required": true
            }
        }
    }
}
```
"edit": {
    "ticket.customer": {
        "shown": true,
        "required": false
    },
    "ticket.agent": {
        "shown": true,
        "required": true
    }
}
}
}

Response:

HTTP Code 201 Created

{"id": 53,
"object_lookup_id": 2,
"name": "sample_integer",
"display": "Sample Integer",
"data_type": "integer",
"data_option": {
    "default": 1234,
    "min": 4,
    "max": 8,
    "null": true,
    "options": {},
    "relation": ""
},
"data_option_new": {},
"editable": true,
"active": true,
"screens": {
    "create_middle": {
        "ticket.customer": {
            "shown": true,
            "required": false,
            "item_class": "column"
        },
        "ticket.agent": {
            "shown": true,
            "required": false,
            "item_class": "column"
        }
    },
    "edit": {
        "ticket.customer": {
            "shown": true,
            "required": false
        }
    }
}

(continues on next page)
"ticket.agent": {
    "shown": true,
    "required": true
},

"to_create": true,
"to_migrate": true,
"to_delete": false,
"to_config": false,
"position": 1550,
"created_by_id": 3,
"updated_by_id": 3,
"created_at": "2021-11-12T18:32:14.213Z",
"updated_at": "2021-11-12T18:32:14.213Z"
}

Payload:

{
    "name": "sample_select",
    "object": "Ticket",
    "display": "Sample Select",
    "active": true,
    "position": 1550,
    "data_type": "select",
    "data_option": {
        "options": {
            "key-one": "First Key",
            "key-two": "Second Key",
            "key-three": "Third Key"
        },
        "default": "key-two",
        "linktemplate": ""
    },
    "screens": {
        "create_middle": {
            "ticket.customer": {
                "shown": true,
                "required": false,
                "item_class": "column"
            },
            "ticket.agent": {
                "shown": true,
                "required": false,
                "item_class": "column"
            }
        },
        "edit": {
            "ticket.customer": {
                "shown": true,
                "required": false
            }
        }
    }
}
Response:

```
HTTP Code: 201 Created

{
  "id": 54,
  "object_lookup_id": 2,
  "name": "sample_select",
  "display": "Sample Select",
  "data_type": "select",
  "data_option": {
    "options": {
      "key-one": "First Key",
      "key-two": "Second Key",
      "key-three": "Third Key"
    },
    "default": "key-two",
    "linktemplate": "",
    "null": true,
    "relation": "",
    "nulloption": true,
    "maxlength": 255
  },
  "data_option_new": {},
  "editable": true,
  "active": true,
  "screens": {
    "create_middle": {
      "ticket.customer": {
        "shown": true,
        "required": false,
        "item_class": "column"
      },
      "ticket.agent": {
        "shown": true,
        "required": false,
        "item_class": "column"
      }
    },
    "edit": {
      "ticket.customer": {
        "shown": true,
        "required": false
      }
    }
  }
}
```

(continues on next page)
Zammad

Payload:

```json
{
    "name": "sample_text",
    "object": "Ticket",
    "display": "Sample Text",
    "active": true,
    "position": 1550,
    "data_type": "input",
    "data_option": {
        "default": "amazing default",
        "type": "text",
        "maxlength": 120,
        "linktemplate": "https://www.google.com/search?q=#{ticket.sample_text}"
    },
    "screens": {
        "create_middle": {
            "ticket.customer": {
                "shown": true,
                "required": false,
                "item_class": "column"
            },
            "ticket.agent": {
                "shown": true,
                "required": false,
                "item_class": "column"
            }
        },
        "edit": {
            "ticket.customer": {
                "shown": true,
                "required": false
            }
        }
    }
}
```

(continues on next page)
"shown": true,
"required": true
}
}
}

**Hint:** Zammad input fields can have 4 different types:

- **email**
- **tel**
- **text**
- **url**

**URL** does not support link-templates

Depending on the chosen input type, Zammad expects different formats of data. E.g.: **email** demands a email address to be provided.

**Response:**

```json
{
    "id": 55,
    "object_lookup_id": 2,
    "name": "sample_text",
    "display": "Sample Text",
    "data_type": "input",
    "data_option": {
        "default": "amazing default",
        "type": "text",
        "maxlength": 120,
        "linktemplate": "https://www.google.com/search?q=#{ticket.sample_text}",
        "null": true,
        "options": {},
        "relation": ""
    },
    "data_option_new": {},
    "editable": true,
    "active": true,
    "screens": {
        "create_middle": {
            "ticket.customer": {
                "shown": true,
                "required": false,
                "item_class": "column"
            },
            "ticket.agent": {
                "shown": true,
                "required": false,
```
Payload:

```json
{
    "name": "sample_treeselect",
    "object": "Ticket",
    "display": "Sample Tree Select",
    "active": true,
    "position": 1550,
    "data_type": "tree_select",
    "data_option": {
        "options": [
            {
                "name": "row one - maximum child depth",
                "value": "row one - maximum child depth",
                "children": [
                    {
                        "name": "row one child level one",
                        "value": "row one - maximum child depth::row one child level one",
                        "children": [
                            {
                                "name": "row one child level two",
                                "value": "row one - maximum child depth::row one child level one::row one child level two",
                                "children": [
                                    {
                                        "name": "row one child level three",
                                        "value": "row one - maximum child depth::row one child level one::row one child level two::row one child level three",
                                    }
                                ]
                            }
                        ]
                    }
                ]
            }
        ]
    }
}
```
"children": [
  {
    "name": "row one child level four",
    "value": "row one - maximum child depth::row one child level one::row one child level two::row one child level three::row one child level four",
    "children": [
      {
        "name": "row one child level five",
        "value": "row one - maximum child depth::row one child level one::row one child level two::row one child level three::row one child level four::row one child level five"
      }
    ]
  }
],
{
  "name": "row two - no childs",
  "value": "row two - no childs"
},
{
  "name": "row three - one child",
  "value": "row three - one child",
  "children": [
    {
      "name": "row three - first and only child",
      "value": "row three - one child::row three - first and only child"
    }
  ]
},
"screens": {
  "create_middle": {
    "ticket.customer": {
      "shown": true,
      "required": false,
      "item_class": "column"
    },
    "ticket.agent": {
      "shown": true,
      "required": false,
      "item_class": "column"
    }
  }
},
(continues on next page)

26.3. Create
Zammad

(continued from previous page)

"edit": {
   "ticket.customer": {
      "shown": true,
      "required": false
   },
   "ticket.agent": {
      "shown": true,
      "required": true
   }
}

Response:

```
HTTP/1.1 201 Created

{  
   "id": 56,
   "object_lookup_id": 2,
   "name": "sample_treeselect",
   "display": "Sample Tree Select",
   "data_type": "tree_select",
   "data_option": {
      "options": [
         {  
            "name": "row one - maximum child depth",
            "value": "row one - maximum child depth",
            "children": [
            {  
               "name": "row one child level one",
               "value": "row one - maximum child depth::row one child level one",
               "children": [
                {  
                   "name": "row one child level two",
                   "value": "row one - maximum child depth::row one child level one::row one child level two",
                   "children": [
                    {  
                       "name": "row one child level three",
                       "value": "row one - maximum child depth::row one child level one::row one child level two::row one child level three",
                       "children": [
                        {  
                           "name": "row one child level four",
                           "value": "row one - maximum child depth::row one child level one::row one child level two::row one child level three::row one child level four",
                           "children": [
                           {  
                              "name": "row one child level five",
                              "value": "row one - maximum child depth::row one child level one::row one child level two::row one child level three::row one child level four::row one child level five"
                           }
                        }
                     ]
                  ]
             }
          ]
       }
    }
 }```
26.3. Create

```json
{ "name": "row two - no childs", "value": "row two - no childs" },
{ "name": "row three - one child", "value": "row three - one child", "children": [
{ "name": "row three - first and only child", "value": "row three - one child::row three - first and only child"
}
], "default": "", "null": true, "relation": "", "nulloption": true, "maxlength": 255 }, "data_option_new": {}, "editable": true, "active": true, "screens": { "create_middle": { "ticket.customer": { "shown": true, "required": false, "item_class": "column" }, "ticket.agent": { "shown": true, "required": false, "item_class": "column" } }, "edit": { "ticket.customer": { "shown": true, "required": false } }
```
Note: Please note that above payloads cover ticket objects. This is fine in most situations, except if you’re looking at the default object permissions. This is why we’re listing these separate for you to view.

The attribute object controls which context is being used:

- Ticket
- User
- Organisation
- Group

Ticket
User
Organization
Group

"screens": {
  "create_middle": {
    "ticket.customer": {
      "shown": true,
      "required": false,
      "item_class": "column"
    },
    "ticket.agent": {
      "shown": true,
      "required": false,
      "item_class": "column"
    }
  },
  "edit": {
    "ticket.customer": {
      "shown": true,
      "required": false
    }
  }
}


```
"screens": {
    "create": {
        "ticket.customer": {
            "shown": true,
            "required": false
        },
        "ticket.agent": {
            "shown": true,
            "required": false
        },
        "admin.user": {
            "shown": true,
            "required": false
        }
    },
    "view": {
        "ticket.customer": {
            "shown": true
        },
        "ticket.agent": {
            "shown": true
        },
        "admin.user": {
            "shown": true
        }
    },
    "edit": {
        "ticket.agent": {
            "shown": true,
            "required": false
        },
        "admin.user": {
            "shown": true,
            "required": false
        }
    },
    "signup": {
        "ticket.customer": {
            "shown": false,
            "required": false
        }
    },
    "invite_customer": {
        "ticket.agent": {
            "shown": true,
            "required": true
        }
    }
}
```
"shown": false,
"required": false
},
"admin.user": {
    "shown": false,
    "required": false
}
},
"invite_agent": {
    "admin.user": {
        "shown": false,
        "required": false
    }
}

"screens": {
    "view": {
        "ticket.customer": {
            "shown": true
        },
        "ticket.agent": {
            "shown": true
        },
        "admin.organization": {
            "shown": true
        }
    },
    "create": {
        "ticket.agent": {
            "shown": true,
            "required": false
        },
        "admin.organization": {
            "shown": true,
            "required": false
        }
    },
    "edit": {
        "ticket.agent": {
            "shown": true,
            "required": false
        },
        "admin.organization": {
            "shown": true,
            "required": false
        }
    }
}
}
"create": {
    "admin.group": {
        "shown": true,
        "required": false
    }
},
"edit": {
    "admin.group": {
        "shown": true,
        "required": false
    }
},
"view": {
    "admin.group": {
        "shown": true
    }
}

### 26.4 Update

Required permission: `admin.object`

**Tip:** Except on the request method, payloads or updating and creating objects are identical. For full payload samples thus scroll up to *Create*.

Zammad will return two attributes during update: `data_option` and `data_option_new`. The first attribute contains the current active values and the second one the new to be values (they’ll become active after executing the database migrations).

**PUT**-Request sent: `/api/v1/object_manager_attributes/{id}`

```json
{
    "id": 50,
    "name": "sample_boolean",
    "object": "Ticket",
    "display": "Sample Boolean",
    "data_type": "boolean",
    "position": 1200,
    "data_option": {
        "options": {
            "true": "yes",
            "false": "no"
        },
        "default": "false"
    }
}
```
Note: Ensure to provide data_option. Zammad is very picky if you leave out this attribute. Please note that changing the object type after creation is not possible.

Response:

```json
HTTP/1.1 200 OK

{
    "name": "sample_boolean",
    "display": "Sample Boolean",
    "data_type": "boolean",
    "position": 1200,
    "data_option_new": {
        "options": {
            "false": "no",
            "true": "yes"
        },
        "default": false,
        "null": true,
        "relation": ""
    },
    "data_option": {
        "options": {
            "false": "very incorrect indeed",
            "true": "very correct indeed"
        },
        "default": null,
        "null": true,
        "relation": ""
    },
    "object_lookup_id": 2,
    "to_config": true,
    "editable": true,
    "id": 50,
    "updated_by_id": 3,
    "active": true,
    "screens": {
        "create_middle": {
            "ticket.customer": {
                "shown": true,
                "required": false,
                "item_class": "column"
            },
            "ticket.agent": {
                "shown": true,
                "required": false,
                "item_class": "column"
            }
        },
        "edit": {
            "ticket.customer": {
                "shown": true,
```
"required": false
    },
    "ticket.agent": {
        "shown": true,
        "required": true
    }
},
"to_create": false,
"to_migrate": false,
"to_delete": false,
"created_by_id": 3,
"created_at": "2021-11-12T18:23:20.883Z",
"updated_at": "2021-11-12T19:30:20.883Z"
}

26.5 Delete

Required permission: admin.object

DELETE-Request sent: /api/v1/object_manager_attributes/{id}

Response:

```
HTTP Code 200 OK
{}
```

26.6 Execute Database Migrations

Required permission: admin.object

**Warning:** After executing the database migrations a restart of Zammad is mandatory. If configured Zammad also can restart automatically (this is the case on Hosted environments) – expect a short downtime.

POST-Request sent: /api/v1/object_manager_attributes_execute_migrations

Response:

```
HTTP Code 200 OK
{}
```
27.1 List

Required permission: user_preferences.access_token

GET-Request sent: /api/v1/user_access_token

Response:

```json
HTTP/1.1 200 OK
Content-Type: application/json

{  
"tokens": [  
{   
"id": 2,   
"user_id": 3,   
"action": "api",   
"label": "test",   
"preferences": {   
   "permission": [    
   "user_preferences.access_token"    
  ]   
},   
"last_used_at": "2021-11-11T14:29:22.765Z",   
"expires_at": null,   
"created_at": "2021-11-10T23:17:46.570Z",   
"updated_at": "2021-11-11T14:29:22.765Z"  
},  
{   
"id": 1,   
"user_id": 3,   
"action": "api",   
"label": "full",   
"preferences": {   
   "permission": [    
   "admin",    
   "ticket.agent"    
  ]   
},   
"last_used_at": "2021-11-10T23:17:46.570Z",   
"expires_at": null,   
"created_at": "2021-11-10T23:17:46.570Z",   
"updated_at": "2021-11-10T23:17:46.570Z"  
]
}
```
"created_at": "2021-11-09T13:17:20.446Z",
"updated_at": "2021-11-10T23:12:06.078Z"
}
,
"permissions": [
{
"id": 1,
"name": "admin",
"note": "Admin Interface",
"preferences": {},
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.544Z",
"updated_at": "2021-11-09T13:12:31.544Z"
},
{
"id": 32,
"name": "admin.api",
"note": "Manage %s",
"preferences": {
"translations": [
"API"
]
},
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.754Z",
"updated_at": "2021-11-09T13:12:31.754Z"
},
{
"id": 26,
"name": "admin.branding",
"note": "Manage %s",
"preferences": {
"translations": [
"Branding"
]
},
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.713Z",
"updated_at": "2021-11-09T13:12:31.713Z"
},
{
"id": 11,
"name": "admin.calendar",
"note": "Manage %s",
"preferences": {
"translations": [
"Calendar"
]
}
]
<table>
<thead>
<tr>
<th>id</th>
<th>name</th>
<th>note</th>
<th>preferences</th>
<th>active</th>
<th>allow_signup</th>
<th>created_at</th>
<th>updated_at</th>
</tr>
</thead>
<tbody>
<tr>
<td>17</td>
<td>admin.channel_formular</td>
<td>Manage %s</td>
<td></td>
<td>true</td>
<td>false</td>
<td>2021-11-09T13:12:31.676Z</td>
<td>2021-11-09T13:12:31.676Z</td>
</tr>
</tbody>
</table>
"translations": [
    "Channel - Formular"
],
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.659Z",
"updated_at": "2021-11-09T13:12:31.659Z"
},
{
  "id": 22,
  "name": "admin.channel_google",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Channel - Google"
    ],
    "active": true,
    "allow_signup": false,
    "created_at": "2021-11-09T13:12:31.687Z",
    "updated_at": "2021-11-09T13:12:31.687Z"
  },
  {
    "id": 23,
    "name": "admin.channel_microsoft365",
    "note": "Manage %s",
    "preferences": {
      "translations": [
        "Channel - Microsoft 365"
      ]
    },
    "active": true,
    "allow_signup": false,
    "created_at": "2021-11-09T13:12:31.693Z",
    "updated_at": "2021-11-09T13:12:31.693Z"
  },
  {
    "id": 24,
    "name": "admin.channel_sms",
    "note": "Manage %s",
    "preferences": {
      "translations": [
        "Channel - SMS"
      ]
    },
    "active": true,
    "allow_signup": false,
    "created_at": "2021-11-09T13:12:31.699Z",
    "updated_at": "2021-11-09T13:12:31.699Z"
  },
  {
    "active": true,
    "allow_signup": false,
    "created_at": "2021-11-09T13:12:31.659Z",
    "updated_at": "2021-11-09T13:12:31.659Z"
  },
  {
    "active": true,
    "allow_signup": false,
    "created_at": "2021-11-09T13:12:31.687Z",
    "updated_at": "2021-11-09T13:12:31.687Z"
  },
  {
    "active": true,
    "allow_signup": false,
    "created_at": "2021-11-09T13:12:31.693Z",
    "updated_at": "2021-11-09T13:12:31.693Z"
  },
  {
    "active": true,
    "allow_signup": false,
    "created_at": "2021-11-09T13:12:31.699Z",
    "updated_at": "2021-11-09T13:12:31.699Z"
  }
}
"id": 21,
"name": "admin.channel_telegram",
"note": "Manage %s",
"preferences": {
    "translations": [
        "Channel - Telegram"
    ]
},
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.682Z",
"updated_at": "2021-11-09T13:12:31.682Z"
},
{
    "id": 19,
    "name": "admin.channel_twitter",
    "note": "Manage %s",
    "preferences": {
        "translations": [
            "Channel - Twitter"
        ]
    },
    "active": true,
    "allow_signup": false,
    "created_at": "2021-11-09T13:12:31.671Z",
    "updated_at": "2021-11-09T13:12:31.671Z"
},
{
    "id": 16,
    "name": "admin.channel_web",
    "note": "Manage %s",
    "preferences": {
        "translations": [
            "Channel - Web"
        ]
    },
    "active": true,
    "allow_signup": false,
    "created_at": "2021-11-09T13:12:31.654Z",
    "updated_at": "2021-11-09T13:12:31.654Z"
},
{
    "id": 40,
    "name": "admin.core_workflow",
    "note": "Manage %s",
    "preferences": {
        "translations": [
            "Core Workflow"
        ]
    },
    "active": true,
    "allow_signup": false,
"created_at": "2021-11-09T13:12:31.807Z",
"updated_at": "2021-11-09T13:12:31.807Z"
},
{
  "id": 36,
  "name": "admin.data_privacy",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Data Privacy"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.783Z",
  "updated_at": "2021-11-09T13:12:31.783Z"
},
{
  "id": 3,
  "name": "admin.group",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Groups"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.565Z",
  "updated_at": "2021-11-09T13:12:31.565Z"
},
{
  "id": 31,
  "name": "admin.integration",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Integrations"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.748Z",
  "updated_at": "2021-11-09T13:12:31.748Z"
},
{
  "id": 59,
  "name": "admin.knowledge_base",
  "note": "Create and setup %s",
  "preferences": {
    "translations": [
      "Knowledge Base"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.748Z",
  "updated_at": "2021-11-09T13:12:31.748Z"
}
[}
  
  ]

},
  
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.992Z",
  "updated_at": "2021-11-09T13:12:31.992Z"
},

{
  "id": 9,
  "name": "admin.macro",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Macros"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.612Z",
  "updated_at": "2021-11-09T13:12:31.612Z"
},

{
  "id": 37,
  "name": "admin.maintenance",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Maintenance"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.789Z",
  "updated_at": "2021-11-09T13:12:31.789Z"
},

{
  "id": 35,
  "name": "admin.monitoring",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Monitoring"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.777Z",
  "updated_at": "2021-11-09T13:12:31.777Z"
},

{
  "id": 33,
  "name": "admin.object",

  (continues on next page)
"note": "Manage %s",
"preferences": {
  "translations": [
    "Objects"
  ]
},
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.760Z",
"updated_at": "2021-11-09T13:12:31.760Z"
},
{
  "id": 5,
  "name": "admin.organization",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Organizations"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.579Z",
  "updated_at": "2021-11-09T13:12:31.579Z"
},
{
  "id": 6,
  "name": "admin.overview",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Overviews"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.591Z",
  "updated_at": "2021-11-09T13:12:31.591Z"
},
{
  "id": 30,
  "name": "admin.package",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Packages"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.738Z",
  "updated_at": "2021-11-09T13:12:31.738Z"
}
{
  "id": 15,
  "name": "admin.report_profile",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Report Profiles"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.648Z",
  "updated_at": "2021-11-09T13:12:31.648Z"
},
{
  "id": 4,
  "name": "admin.role",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Roles"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.572Z",
  "updated_at": "2021-11-09T13:12:31.572Z"
},
{
  "id": 14,
  "name": "admin.scheduler",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Scheduler"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.642Z",
  "updated_at": "2021-11-09T13:12:31.642Z"
},
{
  "id": 28,
  "name": "admin.security",
  "note": "Manage %s Settings",
  "preferences": {
    "translations": [
      "Security"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.642Z",
  "updated_at": "2021-11-09T13:12:31.642Z"
},
{
  "id": 313,
  "name": "admin.queue",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Queue"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.642Z",
  "updated_at": "2021-11-09T13:12:31.642Z"
},
{
  "id": 324,
  "name": "admin.push",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Push"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.642Z",
  "updated_at": "2021-11-09T13:12:31.642Z"
},
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.725Z",
"updated_at": "2021-11-09T13:12:31.725Z"
},
{
"id": 38,
"name": "admin.session",
"note": "Manage %s",
"preferences": {
    "translations": [
        "Sessions"
    ]
},
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.795Z",
"updated_at": "2021-11-09T13:12:31.795Z"
},
{
"id": 27,
"name": "admin.setting_system",
"note": "Manage %s Settings",
"preferences": {
    "translations": [
        "System"
    ]
},
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.719Z",
"updated_at": "2021-11-09T13:12:31.719Z"
},
{
"id": 12,
"name": "admin.sla",
"note": "Manage %s",
"preferences": {
    "translations": [
        "SLA"
    ]
},
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.631Z",
"updated_at": "2021-11-09T13:12:31.631Z"
},
{
"id": 10,
"name": "admin.tag",
"note": "Manage %s",
"preferences": {
    "translations": [
        "Tag"
    ]
},
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.631Z",
"updated_at": "2021-11-09T13:12:31.631Z"
},
{
"id": 30,
"name": "admin.alert",
"note": "Manage %s",
"preferences": {
    "translations": [
        "Alert"
    ]
},
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.631Z",
"updated_at": "2021-11-09T13:12:31.631Z"
}
"translations": [
  "Tags"
],
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.619Z",
"updated_at": "2021-11-09T13:12:31.619Z"
},
{
  "id": 7,
  "name": "admin.text_module",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Text Modules"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.598Z",
  "updated_at": "2021-11-09T13:12:31.598Z"
},
{
  "id": 29,
  "name": "admin.ticket",
  "note": "Manage %s Settings",
  "preferences": {
    "translations": [
      "Ticket"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.731Z",
  "updated_at": "2021-11-09T13:12:31.731Z"
},
{
  "id": 8,
  "name": "admin.time_accounting",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Time Accounting"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.606Z",
  "updated_at": "2021-11-09T13:12:31.606Z"
},
{

(continues on next page)
"id": 34,
"name": "admin.translation",
"note": "Manage %s",
"preferences": {
  "translations": [
    "Translations"
  ]
},
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.767Z",
"updated_at": "2021-11-09T13:12:31.767Z"
},
{
  "id": 13,
  "name": "admin.trigger",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Triggers"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.637Z",
  "updated_at": "2021-11-09T13:12:31.637Z"
},
{
  "id": 2,
  "name": "admin.user",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Users"
    ]
  },
  "active": true,
  "allow_signup": false,
  "created_at": "2021-11-09T13:12:31.556Z",
  "updated_at": "2021-11-09T13:12:31.556Z"
},
{
  "id": 39,
  "name": "admin.webhook",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Webhooks"
    ]
  },
  "active": true,
  "allow_signup": false,
"created_at": "2021-11-09T13:12:31.801Z",
"updated_at": "2021-11-09T13:12:31.801Z"
},
{
"id": 55,
"name": "chat",
"note": "Access to %s",
"preferences": {
  "translations": ["Chat"
  ],
  "disabled": true
},
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.901Z",
"updated_at": "2021-11-09T13:12:31.901Z"
},
{
"id": 56,
"name": "chat.agent",
"note": "Access to %s",
"preferences": {
  "translations": ["Chat"
  ]
},
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.909Z",
"updated_at": "2021-11-09T13:12:31.909Z"
},
{
"id": 57,
"name": "cti",
"note": "CTI",
"preferences": {
  "disabled": true
},
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.916Z",
"updated_at": "2021-11-09T13:12:31.916Z"
},
{
"id": 58,
"name": "cti.agent",
"note": "Access to %s",
"preferences": {
  "translations": ["CTI"
  ]
}
}
```json
{
    "active": true,
    "allow_signup": false,
    "created_at": "2021-11-09T13:12:31.922Z",
    "updated_at": "2021-11-09T13:12:31.922Z"
},
{
    "id": 60,
    "name": "knowledge_base",
    "note": "Manage %s",
    "preferences": {
        "translations": [
            "Knowledge Base"
        ],
        "disabled": true
    },
    "active": true,
    "allow_signup": false,
    "created_at": "2021-11-09T13:12:31.999Z",
    "updated_at": "2021-11-09T13:12:31.999Z"
},
{
    "id": 61,
    "name": "knowledge_base.editor",
    "note": "Manage %s",
    "preferences": {
        "translations": [
            "Knowledge Base Editor"
        ]
    },
    "active": true,
    "allow_signup": false,
    "created_at": "2021-11-09T13:12:32.006Z",
    "updated_at": "2021-11-09T13:12:32.006Z"
},
{
    "id": 62,
    "name": "knowledge_base.reader",
    "note": "Manage %s",
    "preferences": {
        "translations": [
            "Knowledge Base Reader"
        ]
    },
    "active": true,
    "allow_signup": false,
    "created_at": "2021-11-09T13:12:32.012Z",
    "updated_at": "2021-11-09T13:12:32.012Z"
},
{
    "id": 51,
    "name": "report",
    "note": "Manage %s",
    "preferences": {
        "translations": [
            "Knowledge Base Reader"
        ]
    },
    "active": true,
    "allow_signup": false,
    "created_at": "2021-11-09T13:12:32.018Z",
    "updated_at": "2021-11-09T13:12:32.018Z"
}
```
"note": "Report Interface",
"preferences": {},
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.875Z",
"updated_at": "2021-11-09T13:12:31.875Z"
},
{
"id": 52,
"name": "ticket",
"note": "Ticket Interface",
"preferences": {
  "disabled": true
},
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.880Z",
"updated_at": "2021-11-09T13:12:31.880Z"
},
{
"id": 53,
"name": "ticket.agent",
"note": "Access to Agent Tickets based on Group Access",
"preferences": {
  "plugin": [
    "groups"
  ]
},
"active": true,
"allow_signup": false,
"created_at": "2021-11-09T13:12:31.888Z",
"updated_at": "2021-11-09T13:12:31.888Z"
},
{
"id": 41,
"name": "user_preferences",
"note": "User Preferences",
"preferences": {},
"active": true,
"allow_signup": true,
"created_at": "2021-11-09T13:12:31.812Z",
"updated_at": "2021-11-09T13:12:31.812Z"
},
{
"id": 44,
"name": "user_preferences.access_token",
"note": "Manage %s",
"preferences": {
  "translations": [
    "Token Access"
  ]
},
"active": true,
"allow_signup": true,
"created_at": "2021-11-09T13:12:31.812Z",
"updated_at": "2021-11-09T13:12:31.812Z"
}
"active": true,
"allow_signup": true,
"created_at": "2021-11-09T13:12:31.829Z",
"updated_at": "2021-11-09T13:12:31.829Z"
},
{
  "id": 48,
  "name": "user_preferences.avatar",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Avatar"
    ]
  },
  "active": true,
  "allow_signup": true,
  "created_at": "2021-11-09T13:12:31.852Z",
  "updated_at": "2021-11-09T13:12:31.852Z"
},
{
  "id": 49,
  "name": "user_preferences.calendar",
  "note": "Access to %s",
  "preferences": {
    "translations": [
      "Calendars"
    ],
    "required": [
      "ticket.agent"
    ]
  },
  "active": true,
  "allow_signup": true,
  "created_at": "2021-11-09T13:12:31.857Z",
  "updated_at": "2021-11-09T13:12:31.857Z"
},
{
  "id": 47,
  "name": "user_preferences.device",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Devices"
    ]
  },
  "active": true,
  "allow_signup": true,
  "created_at": "2021-11-09T13:12:31.847Z",
  "updated_at": "2021-11-09T13:12:31.847Z"
},
{
  "id": 45,
"name": "user_preferences.language",
"note": "Change %s",
"preferences": {
  "translations": [
    "Language"
  ],
  "active": true,
  "allow_signup": true,
  "created_at": "2021-11-09T13:12:31.834Z",
  "updated_at": "2021-11-09T13:12:31.834Z"
},

{  "id": 46,
  "name": "user_preferences.linked_accounts",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Linked Accounts"
    ],
    "active": true,
    "allow_signup": true,
    "created_at": "2021-11-09T13:12:31.840Z",
    "updated_at": "2021-11-09T13:12:31.840Z"
  },

{  "id": 43,
  "name": "user_preferences.notifications",
  "note": "Manage %s",
  "preferences": {
    "translations": [
      "Notifications"
    ],
    "required": ["ticket.agent"
  ],
    "active": true,
    "allow_signup": true,
    "created_at": "2021-11-09T13:12:31.823Z",
    "updated_at": "2021-11-09T13:12:31.823Z"
  },

{  "id": 50,
  "name": "user_preferences.out_of_office",
  "note": "Change %s",
  "preferences": {
    "translations": [
      "Out of Office"
    ],
    "required": [  
}


```json
{
  "ticket.agent"
}
,
{
  "active": true,
  "allow_signup": true,
  "created_at": "2021-11-09T13:12:31.868Z",
  "updated_at": "2021-11-09T13:12:31.868Z"
}
,
{
  "id": 42,
  "name": "user_preferences.password",
  "note": "Change %s",
  "preferences": {
    "translations": [
      "Password"
    ]
  },
  "active": true,
  "allow_signup": true,
  "created_at": "2021-11-09T13:12:31.818Z",
  "updated_at": "2021-11-09T13:12:31.818Z"
}
}
```
27.3 Delete

Required permission: user_preferences.access_token

DELETE-Request sent: /api/v1/user_access_token/{id}

Response:

```
HTTP/1.1 200 OK
{}
```
Zammad comes with a collection of scripts for easy backup & restore for default installations. These scripts are located within /opt/zammad/contrib/backup.

**Note:** Important things to note before hand

These scripts do not come with any warranty and may not work in your specific use case. This depends on the configuration and installation type of your instance.

You should always regularly test and review the functionality! If the script functionality or scope is not working for your cases, feel free to copy these to a independent location and adjust the scripts as needed.

### 28.1 Getting Started

#### 28.1.1 Backup configuration

Before you can run either a backup or restoration, the scripts requires you to provide a configuration file. We’re shipping a `config.dist` within the `/opt/zammad/contrib/backup` directory which you can simply rename.

To do so run the following commands as either root or zammad user.

```
$ cd /opt/zammad/contrib/backup/
$ mv config.dist config
```

If below default values are not working for you or your installation in general, this is the best moment to adjust the configuration file as needed.

After this you’ll be ready to continue with either *creating your first backup* or *restoring an existing backup*.

**BACKUP_DIR** Default: `/var/tmp/zammad_backup`

Tell the backup script where to write your backup files to.

**Warning:** Ensure that the user you’re going to use for backing up Zammad (either root or zammad by default) has enough permissions to write into the target directory structure.

In case the directory is not available yet, the backup script will attempt to create the directory.
**Hint:** Ensure to have enough space available on the backup location. Zammad always creates full backups. While we do compress backups, expect worst case ratios of 1 (no compression at all) depending on your attachments!

**HOLD_DAYS** Default: 10

How many days should the backup script keep old backups? This value contains 60 minutes grace period (so e.g. 10 days plus 1 hour) for safety reasons.

Old backups are removed *before* creating the actual (current) backup.

**Note:**

0 will keep the last 25 hours worth of backup
-1 will always remove all available backups (aka only keep current backup)

**FULL_FS_DUMP** Default: yes (accepts: yes or no)

**Note:** If you can’t decide, our clear suggestion is setting this to no.

Setting this option to no allows you to only backup usage data without any environmental files from your old host. This allows you to backup your Zammad database together with the attachments you’ve stored within the file system.

Please refer Storage Settings to learn how to change the storage location of your attachments.

**DEBUG** Default: no (accepts: yes or no)

Having issues and want to fiddle around? Setting this option to yes may help you with this. It contains useful debug messages at strategic points.

**Warning:** This option potentially returns sensitive information to standard out! Do not use this option in productive environments or ensure to turn it off after testing.

### 28.1.2 Create Backup

```
# Zammad backup started - Fri Jan 21 17:53:44 CET 2022!
creating file backup...
  ... as full dump
creating postgresql backup...
Ensuring dump permissions ...

# Zammad backedup successfully - Fri Jan 21 17:53:57 CET 2022!
```

Sample backup process with default settings.

**Note:** Before running your first backup, please have a look at Backup configuration.
In general, running a Zammad backup is as simple as running

```
$ /opt/zammad/contrib/backup/zammad_backup.sh
```

Please make sure to test the backup function manually with the user you’re planning the backup first. This ensures that your backup really is running as expected.

**Remarks**

The backup script can be either run as `zammad` or `root` user.

Stopping Zammad is not required (but suggested) technically, but may be in your use case!

(Keep in mind that a running Zammad instance keeps changing data which may be an issue during long backup runs)

**Hint:** Having trouble backing up?

Have a look at the troubleshooting section to address your issues.

### 28.1.3 Restore

```
# Zammad restore started - Fri Jan 21 17:54:13 CET 2022!

The restore will delete your current database!
Be sure to have a backup available!

Please ensure to have twice the storage of the uncompressed backup size!

Note that the restoration USUALLY requires root permissions as services are stopped!

Enter 'yes' if you want to proceed!
Restore?: yes
Enter file date to restore:
20220120124714
20220121175344
File date: 20220121175344
Enter db date to restore:
20220120124714
20220121175344
DB date: 20220121175344
# Stopping Zammad
# Checking requirements
# ... Dropping current database zammad
Dropped database 'zammad'
# ... Creating database zammad for owner zammad
CREATE DATABASE
# Restoring PostgreSQL DB
# Restoring Files
# Ensuring correct file permissions ...
# Clearing Cache ...
# Starting Zammad
```

(continues on next page)
Zammad restored successfully - Fri Jan 21 17:54:34 CET 2022!

Sample backup process.

Hint: Migrating from host to host…?
Please refer Migrate Zammad to new host for more additional information you may want to consider during a migration.

Warning:
This documentation page expects a fully installed Zammad version.
It also expects you to restore Zammad on the same host and version!

Before restoring your backup, please note the following:

• The restoration process stops & restarts the Zammad service. This means you usually have to run the restoration script as root user.

  Note:
  - This is mandatory for package installations
  - On Source code installation this does not work because of different environments - you could load it before hand as root user to have access to Zammad specific commands.
  - If both approaches above do not fit for your case, consider adjusting the backup and restore scripts to your need in an independent directory. You’re working out of script and documentation scope!

• PostgreSQL based installations will drop and re-create the database! MySQL / MariaDB based installations restore on the existing database.

• You require at least twice the backed up Zammad instance size of free storage. If you have the dump only, factor 3 could be a good number.

Step 1: Copy your backup files to a fitting location (if needed) This basically is a given usually if you run a normal restore. Ensure that the user you’re using for restoration is allowed to read the backup files - writing is required for /opt/zammad/.

  Hint: So many files, what’s my backup files?
  The Zammad backup consists of two files. This is their format:

  <timestamp>_zammad_db.psql.gz
  <timestamp>_zammad_files.tar.gz

  There’s also two symlinks in your backup directory showing to the newest backup created.

Step 2: Configure the backup script (if needed)
On new installation it’s required. For restoration this mainly affects the backup file location.
Please consult Backup configuration for more.
Step 3: Run the restore  Restoration works via two possible ways, depending on how interactive you want to go.


**Warning:** Restoring old backups may overwrite your `database.yml`. You can find out if that’s the case by having a look into the file `.tar.gz` within the `config` directory. If you can see a `database.yml` there, ensure to save the original version before restoring.

If you found the trap already, you can try the `Database Helper: (re)set password`.

Interactive restoration (recommended)

Non-interactive restoration

```
$ /opt/zammad/contrib/backup/zammad_restore.sh
```

Provide the requested information to the script wait for the restoration to finish. Depending on the size of your backup and host performance this may take some time.

**Warning:** Only use the following option if you know what you’re doing! The following command will overwrite existing data without further prompts!

```
# When called with a timestamp argument (matching the backups filename),
# Zammad will proceed immediately to restoring the specified backup.
$ /opt/zammad/contrib/backup/zammad_restore.sh 20170507121848
```

**Hint:** Having trouble restoring?

Have a look at the `troubleshooting section` to address your issues.

Step 4: Re-install Zammad if restoring a full filesystem restore  Zammad’s backup scripts backup the whole filesystem of Zammad. This is mainly for backward compatibility but not a hard requirement.

If your filesystem dump contains attachments only (the tar will contain a `storage` folder only) skip this step!

For a better overview, please see: step 9 of our migration path.

Step 5: Apply missing environmental settings

**Note:** This does not apply to Docker images, as the following settings should be applied upon every start automatically.

If you’ve set any environmental settings like higher web concurrency due to required `Performance Tuning`, please re-apply your settings now.

If not already done, please install Elasticsearch now (if you want to use it). Follow Step 3: Connect Zammad to reconfigure your installation for Elasticsearch use and rebuild the search index.

You are now ready to continue your work. The rebuild of your search index can safely run during your work, but will cause a degraded search performance and may lead to temporarily not found data.
28.1.4 Migrate Zammad to new host

This is a proof of concept, not a full how to. Your environment may be different. Please note that the steps described on this page are an addition to backing up and restoration.

They’re not meant to stand alone - we’ll link and note this at the relevant parts.

If anything goes wrong, please consult the Zammad Community or consider paid support options.

---

**Hint:** Migrating from Zammad SaaS? Skip to step 7.

**Step 1:** Note down your environmental adjustments This mainly affects performance tuning settings. This will be important after restoring.

**Step 2:** Install Zammad on the destination host For the easiest restoration path possible, please install the same version like your origin instance. You could also consider updating the old instance before migrating.

Choose between these installation types:

- **package**
- **source code**
- **docker-compose**

**Warning:** Restoration & Migration on docker based installation may differ. While the steps are the same on most parts, it is not covered by this documentation!

**Step 3:** Activate maintenance mode This ends agents and customers sessions. Learn more about the maintenance mode in Zammad.

**Step 4:** Disable your communication channels This is just a safety measurement. As our restore scripts starts Zammad automatically, this may help if something is not in a correct state.

**Step 5:** Stop and disable Zammad Make sure to no longer have Zammad change data before backing up.

```
$ systemctl disable zammad
$ systemctl stop zammad
```

**Note:** This does not apply to docker based environments.

**Step 6:** Backup! Follow our documentation part for backup creation.

**Hint:** Note down if you’ve created a full filesystem dump or only backed up your attachments. This will be important for the restoration.

If you want to go with the easiest way, consider only dumping your attachments. Learn more on our configuration page.

**Step 7:** Transfer your backup files You’ll find the backup location within the `conf` file on the backup directory. Make sure to adjust the backup configuration on the destination host according to our configuration page to provide the correct backup file directory.

Provide the file location you transferred the backup files to.
Step 8: Restore your backup  Follow the steps 1 to 3 of our restoration page to restore the backup on the new host.

**Warning:** If you’re running a source code installation, install the same version before hand. This reduces environment fiddlings a lot.

If you don’t want that, you can find a version list on the Software page.

**Warning:** Restoring old backups may overwrite your database.yml. You can find out if that’s the case by having a look into the file .tar.gz within the config directory. If you can see a database.yml there, ensure to save the original version before restoring.

If you found the trap already, you can try the Database Helper: (re)set password.

**Important**  Stop Zammad after the restoration has finished.

If you experience issues during restoration, please consult Troubleshooting Backup & Restore.

Step 9: Run required maintenance tasks after restoring  After successful restoration, please continue below depending if you’ve only backed up your attachments or had a full filesystem dump.

**Tip:** Migrating from Zammad SaaS? You’ve received an attachment dump!

Attachment dump (recommended)

Full filesystem dump

**Note:**  Keep in mind that docker-compose and source code installations do not know zammad run. Below commands show the package installation way, just remove all zammad run parts from the commands and run them.

This means: zammad run rails c would be rails c.

Step 9.1: Clear the cache

$ zammad run rails r "Cache.clear"

**Tip:** Skip steps 9.1, and 9.2, and 9.3 if you do not have the last possible Zammad version installed. However, make sure to run the next steps in the following order: step 12, then step 10, then step 11.

**Note:**

This step is only needed, if one of the following points is met:

- The source and destination Zammad-Version are not the same
- The Zammad-installation is not a source code installation
- The Zammad-Backup is not an Export from Hosted-Setup

Full-Dumps for source code installations are not covered, however, basically the same below applies to you: You have to ensure that the environments and application files are overwritten with the new / correct version.
Zammad files are distribution and version specific!

Step 9.1: Uninstall and reinstall Zammad without resolving dependencies

**Debian, Ubuntu**

```
$ dpkg -r --force-depends zammad
$ apt install zammad
```

**OpenSUSE**

```
$ zypper remove -R zammad
$ zypper install zammad
```

**Hint:** You’re unsure if above is really required and a mere reinstall would be enough? If you run a dedicated install command on for Zammad and receive the following, you absolutely have to run above to fix your installation.

```
$ root@zammad:/# apt-get update && apt install zammad
Reading package lists... Done
Building dependency tree
Reading state information... Done
zammad is already the newest version (x.x.x-xxxxxx.xxxxxx.xxx).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

Step 9.2: Clear the cache

**Note:** Keep in mind that docker-compose and source code installations do not know `zammad run`. Below commands show the package installation way, just remove all `zammad run` parts from the commands and run them.

This means: `zammad run rails c` would be `rails c`.

```
$ zammad run rails r "Cache.clear"
```

Step 9.3: Ensure Zammad is running

```
$ systemctl status zammad
# If Zammad is not running, run below
$ systemctl start zammad
```

Step 10: Apply missing environmental settings

**Note:** This does not apply to Docker images, as the following settings should be applied upon every start automatically.

If you’ve set any environmental settings like higher web concurrency due to required *Performance Tuning*, please re-apply your settings now.

If not already done, please install Elasticsearch now (if you want to use it). Follow *Step 3: Connect Zammad* to reconfigure your installation for Elasticsearch use and rebuild the search index.
You are now ready to continue your work. The rebuild of your search index can safely run during your work, but will cause a degraded search performance and may lead to temporarily not found data.

**Step 11: Re-enable Channels and deactivate maintenance mode**  Set the previous deactivated channels back to active if you’re sure everything was successful. At this point Zammad will start to change data!

After verifying the functionality of your channels, allow your agents and customers back in by disabling the maintenance mode.

Learn more about the maintenance mode in Zammad.

**Step 12 (optional): Update Zammad to latest possible version**  In case the backup source was not on the latest possible version, please update your Zammad installation now.

In case your installed version is fairly old, please note the upgrade path notes on our updating zammad page.

### 28.1.5 Troubleshooting Backup & Restore

Errors have been made, possibly they can be corrected through.

**Exit codes**

Backup & Restore script come with exit codes to help you within possible autonomous handling. However, we do not guarantee a complete error handling.

Beside the exit codes, there’s also error messages returned to standard out.

<table>
<thead>
<tr>
<th>Code</th>
<th>Description / Situation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>The script finished successfully (or the error is not handled).</td>
</tr>
<tr>
<td>1</td>
<td>This is a general error. Most often used for script aborts due to incorrect information provided or information missing.</td>
</tr>
<tr>
<td>2</td>
<td>There was an error with database handling. This usually either happens if your database server does not meet script requirements, login data being invalid or „broken database dumps.</td>
</tr>
<tr>
<td>3</td>
<td>There were issues with file / folder permissions.</td>
</tr>
</tbody>
</table>

**Classics**

Here’s some classics you may encounter.

**Hint:**  Your issue is not listed...?

Please consult the Zammad Community for technical assistance.

**password authentication failed**-or-**peer authentication failed**  This indicates that the password of your Zammad DB user is either different from your database.yml or the wrong database server may be contacted.

„But my Zammad instance is running, how can it be wrong?“  Zammad may fall back to socket connection which is why you didn’t notice.

**What to do ...**

- Ensure that the provided user credentials are correct
Tip: Lazy users can use our helper

In the backup directory you’ll find a Database Helper: (re)set password.

Ident authentication failed for user This indicates your database server does require ident authentication.

That authentication method is not supported by our scripts.

What to do … Check pg_hba.conf of your PostgreSQL-Server and adjust it if needed.

Usually authentication can be allowed like so:

<table>
<thead>
<tr>
<th># THIS IS A SAMPLE AND MAY NOT FIT YOUR ENVIRONMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>host  all  all  127.0.0.1/32  md5</td>
</tr>
<tr>
<td>host  all  all  ::1/128    md5</td>
</tr>
</tbody>
</table>

Please consult the official PostgreSQL documentation for this, as this is out of our documentation scope.

**WARNING:** You don’t seem to have any attachments in the file system! This indicates you’ve set FULL_FS_DUMP to no but your instance currently does not save attachments to file system.

This warning will be shown once before creating an empty directory to allow the backup process to continue successfully.

If you believe that this is an error, please see Storage Settings. In case the issue persists, please consult the Zammad Community.

### 28.1.6 Helper scripts

*Danger:* The following scripts are potentially destructive

You should never run scripts which scopes you don’t understand. Below scripts potentially can make things worse which is why you should evaluate them before hand.

You’re running these scripts at your own risk.

If we found a script is helping you more than 30 lines of new documentation, we may have added a helper script.

#### Database Helper: (re)set password

**Limitations**

- This script is working for PostgreSQL installations only.
- Only local database servers are supported (script changes user).
- This script requires to be run as root or similar privileged user!

**Scopes** Mostly the following installation types will be affected / relevant:

- package installations (especially CentOS & SuSe)
- possibly source code installations

**Functionalities** The script will do the following actions depending on the situation automatically for you.

It will double tab by asking for your confirmation up front.
• If `database.yml` contains an empty password line, a new password will be generated, and set for the database user of Zammad, and saved to the configuration file.

• If `database.yml` contains a password, it will be used to set the password of the Zammad database user.

**Warning:** **Downtime**
The script will automatically stop and start Zammad!

**Usage** Run `/opt/zammad/contrib/backup/zammad_db_user_helper.sh` and follow the instructions. No specific configurations are required.

If errors occur the script will try to bring Zammad back online before exiting. Please ensure that your service is running.

Before you continue, please also note the listed limitations to save your precious time.

**Adjust script settings** Learn more about configuration options for backup and restore to see scopes better.

**Create Backups** How to create full dumps of your Zammad installation.

**Restore Backups** Update went berserk and you need to go back? How to restore your instance on a new or the same host.

**Migrating to new hosts** This is a general summary on how to best migrate Zammad from host to host. We’ll reference to backup creation and restoration as needed.

**Troubleshooting** Things hit the fan? This page might help you out of that pit.

**Helper scripts** These scripts may be helpful if Backup & Restore does not work as expected. However note that these are potentially destructive.

## 28.2 Limitations

Please note the following limitations which may affect script functionality or availability.

• Restoration via script on docker and source code based installations may not work and is out of scope of this documentation as of now

• Backup & Restore is only available for PostgreSQL and MySQL / MariaDB like installations

• Switching / Converting database installations is not possible

---

**Note:** If you require support with migrating your MySQL / MariaDB installation into a PostgreSQL installation, you can contact Zammad's sales team for commercial support.

---

• Starting with Zammad 5.0 the scripts require user & password authentication. This is supported by most of our installation types

• Backup & Restore is always a full dump of everything (no incrementals)

• Restoring or backing up specific information (e.g. Tickets, Users, . . .) is not supported

• Environmental settings (like e.g. Configuration via Environment Variables) are not backed up and thus require you to manually set them on a new host

• Restoration into a older Zammad version is not possible nor supported
• *Do not* attempt to restore backup files from custom scripts with the provided scripts by Zammad. This is most likely subject to fail or bring issues you may discover too late.
Use these environment variables to configure Zammad’s behavior at runtime.

**Note:** What’s an environment variable, and how do I “use” it?

Unfortunately, that question has a very long answer that goes beyond the scope of this article. How you set environment variables will depend on how you installed Zammad (e.g., source, package, or Docker).

But for package installations, here’s a short answer:

```bash
# set OPTION to "value"
$ zammad config:set OPTION=value
$ systemctl restart zammad

# get OPTION
$ zammad config:get OPTION

# unset OPTION
$ zammad config:unset OPTION
$ systemctl restart zammad
```

To learn more, do some googling on environment variables and the shell environment (or execution environment) in Unix.

### 29.1 General Options

**APP_RESTART_CMD** The command Zammad will use to automatically restart the server after changes have been made in the Object Manager. (E.g., "systemctl restart zammad")

If this is undefined, you will have to restart manually after making changes in the Object Manager.

Default: unset

**RAILS_LOG_TO_STDOUT** Print output directly to standard output instead of /var/log/zammad/production.log.

**Warning:** On package installations, this setting can be overwritten during update.

Use enabled to turn this option on only until the next update. Use true to turn it on permanently.

Default: unset
29.2 Network Options

**ZAMMAD_BIND_IP** The IP address that the web server is bound to.

Default: 0.0.0.0

**ZAMMAD_RAILS_PORT** The port that the web server is exposed on.

Default: 3000

**ZAMMAD_WEBSOCKET_PORT** The port that the web socket server is exposed on.

Default: 6042

*Note:* Remember to update your web server config to reflect any changes you make here.

29.3 Performance Tuning

**Warning:** Each of these settings comes with its own tradeoffs.
There are no “recommended values” here; the optimal configuration will depend on your system’s resources and typical application load.

Proceed with caution: when adjusting any of these settings, there is a point at which performance will begin to degrade rather than improve, or other problems will begin to crop up.

**Tip:** How can I find out how many users are currently on Zammad?

```
$ zammad run rails r "p Sessions.list.uniq.count"
```

**WEB_CONCURRENCY** How many instances of the application server to keep open at a time.

Increasing this can reduce loading times when too many users are on Zammad at once.

Default: unset

**ZAMMAD_SESSION_JOBS_CONCURRENT** How many instances of the session worker to run at a time.

Increasing this can speed up background jobs (like the scheduler) when too many users are on Zammad at once.

Generally speaking, it should only be useful to adjust this setting if you have more than 40 active users at a time.

**Warning:** Session workers can be extremely CPU-intensive.

In some cases, they can reach 100% CPU utilization on their own. Increasing this setting is safer on systems with more cores.

Default: unset

**ZAMMAD_PROCESS_SCHEDULED_JOBS WORKERS** Allows spawning an independent process just for processing scheduled jobs like LDAP syncs. This can free up Zammad’s background worker for other tasks when running tasks that require fairly long.
Zammad

Default: unset
Maximum number of workers: 1

**Danger:** Disable processing of scheduled jobs by setting `ZAMMAD_PROCESS_SCHEDULED_JOBS_DISABLE`. Doing so on productive instances will draw important parts of your instance not working. **WE STRONGLY encourage against using this flag.**

**ZAMMAD_PROCESS_DELAYED_JOBS_WORKERS** How many processes should work on delayed jobs? Increasing this can improve issues with delayed jobs stacking up in your system. You may want to try to use `ZAMMAD_SESSION_JOBS_CONCURRENT` before though.

Default: unset
Maximum number of workers: 16

**Warning:** This option can be *very* CPU-intensive.

**Danger:** Disable processing of delayed jobs by setting `ZAMMAD_PROCESS_DELAYED_JOBS_DISABLE`. Doing so on productive instances will draw important parts of your instance not working. **WE STRONGLY encourage against using this flag.**

**Warning:** Above settings *may* consume all available database connections. Please consider the database server configuration section for more.

**Note:** The options listed below allow you to distribute Zammad processes over several application nodes. Even if that’s not your goal, they may provide great benefits on bigger installations.

Please note that distribution of processes on several nodes is out of the scope of this documentation for various reasons.

**REDIS_URL**

Store your web socket connection information within Redis.
To do so, tell Zammad where to find your Redis instance: `redis://your.redis.server:6379`.
If not provided, Zammad falls back to file system (`/opt/zammad/tmp/websocket_*`).
Default: unset

**MEMCACHE_SERVERS**

Store your application cache files within Memcached.
To do so, tell Zammad where to find your Memcached instance: `your.memcached.server:11211`.
If not provided, Zammad falls back to file system (`/opt/zammad/tmp/cache*`).

29.3. Performance Tuning 339
Tip: Size may be important

Memcached allows you to restrict the maximum size Zammad may store as cache. This comes in handy in terms of performance and keeping caching files small. 1 GB should be a reasonable size.
CHAPTER THIRTY

CONFIGURE DATABASE SERVER

Note: Parts of this page also applies to both supported database servers. We can't provide a complete how to and will only enlighten the relevant parts for Zammad.

Within database.yml (config/ directory) you can define the allowed pool size. By default each Zammad process takes up to 50 connections (pool: 50).

This should be fairly enough for every use case. If you experience database connection timeouts or similar pool errors, this usually indicates to other issues that are relevant to your PostgreSQL.

Note: Below only affects PostgreSQL-Servers. All relevant steps for MySQL are mentioned on Software because they're relevant before installation.

Below you can the locations of the relevant PostgreSQL configuration files to adjust. Keep in mind that versions may differ from your setup - adapt where needed.

Ubuntu / Debian
CentOS / OpenSUSE
other

/etc/postgresql/(10|11|12)/main/postgresql.conf
/var/lib/pgsql/data/postgresql.conf

Can't find your configuration files? You can run the following command to get the path:

$ sudo -u postgres psql -c 'SHOW config_file'

Adjust max_connections (mandatory) Zammad will take up to 200 connections by default, with below command you can raise this limit fairly high.

# Raise maximum allowed number of connections
$ sed -i "/max_connections/c\max_connections = 2000" <postgresql-configuration-file>

# Apply changes by restarting postgresql and Zammad (in this order)
$ systemctl restart postgresql zammad

Adjust PostgreSQL for bigger instances (optional)
**Warning:** Check below settings first and ensure your system is able to provide the requirements! Below settings are what we found to be useful, everything else is out of scope of this documentation!

```bash
# Caching improvements
$ sed -i "/shared_buffers/c\shared_buffers = 2GB" <postgresql-configuration-file>
$ sed -i "/temp_buffers/c\temp_buffers = 256MB" <postgresql-configuration-file>
$ sed -i "/work_mem/c\work_mem = 10MB" <postgresql-configuration-file>
$ sed -i "/max_stack_depth/c\max_stack_depth = 5MB" <postgresql-configuration-file>

# Apply changes by restarting postgresql and Zammad (in this order)
$ systemctl restart postgresql zammad
```
PRIVACY & DATA RETENTION

How long does Zammad hold onto user data? How can I manage its user data retention behavior?

31.1 On-Premises Data

The following kinds of data are stored locally on the production system:

Tickets and users By default, Zammad never automatically deletes tickets or users.

To enable automatic deletion of tickets after a given interval, use the scheduler. To manually delete users and all their associated tickets (e.g., in compliance with a “Right to Forget” request under the GDPR), use the console.

Note: The ability to delete users via the admin panel is planned for a future release of Zammad.

Chat sessions Once a chat session has been marked closed, it is scheduled for automatic deletion 3 months later.

IP address logs for chat sessions can be manually deleted by following the directions here.

CTI caller log The caller log shows only the 60 most recent entries. Each entry in the caller log is automatically deleted after 12 months.

Log files Zammad writes log files to disk (typically under /opt/zammad/log/).

Package installations will set up a separate system utility called logrotate to rename and archive (or rotate) log files on a nightly basis and remove old logs after 14 days.

If installing from source, it is strongly recommended to configure logrotate or a similar log management utility; Zammad will not purge old logs on its own.

User sessions Zammad maintains session information about every user currently logged in.

This information is automatically purged when a user logs out, and can be viewed or manually deleted via the admin panel (under System → Sessions). Users may also delete their own session information via the user preferences menu, under Device.

Session information includes IP address (and possibly geographic location), browser, time of original login, and time of last visit.
31.2 External Services

Zammad utilizes third-party web services for certain functions, meaning that user data may occasionally be sent or exposed to third parties. These functions can be individually disabled in the admin panel under Settings → System → Services.

Note: By default, the third-party services that Zammad relies on are mostly ones hosted and managed by the Zammad Foundation itself, but Zammad can be extended to interface with other services instead.

The source code for these third-party service integrations can be found here.

**Images** No private images or personally-identifying information are stored on images.zammad.com.

The Images service caches publicly-available images from sources like Gravatar and serves them to the Zammad application as user avatars and organization logos. These images are discovered using MD5 digests of user email addresses and organization domain names. User avatars are cached for 7 days; organization logos are cached for 30 days.

**GeoCalendar** No user information is stored or cached on geo.zammad.com.

As part of its service-level agreement (SLA) functionality, Zammad requires detailed, localized calendar information (e.g., to set the time zone and accommodate national holidays and daylight savings time). The GeoCalendar service is used to retrieve this information.

**GeoIP** No user information is stored or cached on geo.zammad.com.

One of Zammad’s security features is to track user sessions based on the user’s browser and country of origin. Suspicious login activity from a different browser or country may trigger Zammad to dispatch an alert email to the affected user. The GeoIP service is used to associate IP addresses to a geographic origin.

**Geolocation** Since Zammad’s geolocation service relies on Google’s Geocoding API, its use is subject to the Google Privacy Policy.

Zammad uses geolocation to associate tickets with locations to support map-style ticket overviews, which display tickets as points on a map rather than items in a list.
This guide will discuss frequently asked questions and how to resolve common problems with Zammad.

Note: Troubleshooting unsuccessful or issue not described?

If you can’t solve your issue using the provided troubleshooting steps or can’t find your particular issue described here, feel free to ask the community for technical assistance.

32.1 Data missing from the Web-UI / Search data missing or incomplete

A commonly reported issue is data missing from the Web-UI. This could be Tickets, Articles, Users or anything else indexed by Elasticsearch and can be caused by missing or incomplete indexes.

If you are experiencing this issue and installed Elasticsearch according to Set up Elasticsearch, please follow these steps to make sure Elasticsearch is working correctly.

Step 1: Verify Elasticsearch is running

```bash
# check elasticsearch status
$ systemctl status elasticsearch
```

Note: This should output something like the following, make sure it says Active: active (running):

```
elasticsearch.service - Elasticsearch
    Loaded: loaded (/lib/systemd/system/elasticsearch.service; enabled; vendor...
-- preset: enabled)
    Active: active (running) since Tue 2021-07-20 09:38:21 UTC; 1h 4min ago
    Docs: https://www.elastic.co
    Main PID: 1790 (java)
```

Otherwise, try starting it and check again:

```bash
# restart elasticsearch and check status
$ systemctl restart elasticsearch
$ systemctl status elasticsearch
```
Warning:
If this fails, your Elasticsearch installation is probably broken.
Try completely purging and reinstalling Elasticsearch according to Set up Elasticsearch

Step 2: Verify the ingest-attachment plugin is installed correctly

# list installed elasticsearch plugins
$ /usr/share/elasticsearch/bin/elasticsearch-plugin list

Note: The output should include ingest-attachment.
Otherwise, try reinstalling the ingest-attachment plugin and check again:

$ /usr/share/elasticsearch/bin/elasticsearch-plugin remove ingest-attachment
$ /usr/share/elasticsearch/bin/elasticsearch-plugin install ingest-attachment
$ systemctl restart elasticsearch
$ /usr/share/elasticsearch/bin/elasticsearch-plugin list

Step 3: Verify Zammad can access Elasticsearch and rebuild the indexes

# force zammad to drop and rebuild the elasticsearch indexes
$ zammad run rake zammad:searchindex:rebuild

Note: This should start rebuilding the indexes and output it’s progress:

Dropping indexes... done.
Deleting pipeline... done.
Creating indexes... done.
Creating pipeline... done.
Reloading data...
  - Chat::Session...
    done in 0 seconds.
  - Cti::Log...
    done in 0 seconds.

Depending on the system performance and amount of data, this can take a while to complete.
Please let this task finish completely and wait until it drops you back to the console.

Warning:
If this fails or throws an error, there might be something else wrong with your installation.
Make sure you followed the complete Elasticsearch set up and integration procedure according to *Set up Elasticsearch.*

**Tip:** In many situations where you’re not successful with above steps, you may want to check Elasticsearch’s log file: `/var/log/elasticsearch/elasticsearch.log`.

After completing these steps, you should have verified your Elasticsearch installation is running and rebuilt the indexes.  
If this does not resolve your issue, please ask the community.
This guide will discuss how to set up single sign-on using Microsoft Active Directory.

Note: SSO can only be configured on self-hosted installations.

Fig. 1: As of Zammad 3.5, enabling SSO adds a new button to the sign-in page.

### 33.1 Conceptual Overview

Like every other web application out there, Zammad has its own logic for signing users up, storing their passwords, authenticating them, and managing their sessions.

If your IT department keeps its own user identity store (like Active Directory), Zammad’s SSO support allows you to leverage that existing auth system so that anyone with an account on your local intranet will 1) automatically have an account in Zammad and 2) be able to log in with a single click.

Note: If you don’t have this IT infrastructure but still want one-click login, see Third-Party Authentication for alternatives.

### 33.1.1 How does it work?

Once enabled, single sign-on activates an endpoint at https://your.zammad.host/auth/sso. When the Zammad server receives a GET request at this endpoint with a valid username in any of the following:

- an X-Forwarded-User request header
- a REMOTE_USER web server environment variable
- an HTTP_REMOTE_USER web server environment variable

it creates a new session for that user.

Note: Wait. SSO allows you to sign in with only a username?

In principle, yes.

How is that okay?
In this guide, we configure our web server (Apache) to intercept all requests to the `/auth/sso` endpoint. Instead of forwarding them to Zammad, Apache initiates a three-sided login process (*Kerberos authentication*) between the itself, the user, and the Active Directory server.

If Active Directory doesn’t recognize the user or their password, Zammad never sees the request, and the session is never created.

**What does this all mean?**

It means there are many ways you could set up SSO—you don’t need to follow this guide or even use Active Directory or Kerberos—but if you don’t know what you’re doing, you’re going to end up with a *massive* security hole.

### 33.2 Getting Started

**Hint:** Too busy to handle it on your own?

We’ve got you covered. Our experts offer custom-tailored workshops to get your team up and running fast and with confidence. *Just drop us a line!*

You will need:

- a Microsoft Active Directory environment with
  - root access
  - support for AES 256-bit encryption
- a Zammad host with
  - root access
  - a fully-qualified domain name (FQDN)
- some familiarity with system administration (*e.g.*, Apache configuration)

For best results, set up LDAP integration to make sure your Active Directory and Zammad user accounts are always in sync.

### 33.3 Step 1: Configure Active Directory

In the Kerberos authentication scheme, the *authentication server* (Active Directory) needs to maintain shared secrets with the *service* (Zammad). To make this possible, we need to register a *service principal name* (SPN) for Zammad on Active Directory.

**Note:** These directions have been confirmed on Windows Server 2016.
33.3.1 1a. Create a service account

You may use an existing service account if you have one. Admin privileges are not required; a normal user account will do.

Fig. 2: Select “This account supports Kerberos AES 256 bit encryption” under Properties > Account > Account options.
33.3.2 1b. Register an SPN for Zammad

Note: Replace the following placeholders in the command below:

- `<zammad-host>` Zammad FQDN
- `<service-acct>` Service account logon name
- `<password>` Password of the service account (Option `/pass * did prove to not work`)
- `<domain>` Windows domain
- `<master-domain-controller>` Master domain controller IP/FQD

Below command will prompt for the users password.

```
$ setspn -s HTTP/<zammad-host> <service-acct>
$ ktpass /princ HTTP/<zammad-host>@<DOMAIN> \\
   /mapuser <service-acct> \\
   /crypto AES256-SHA1 \\
   /ptype KRB5_NT_PRINCIPAL \\
   /pass <password> -SetPass +DumpSalt \\
   /target <master-domain-controller> \\
   /out zammad.keytab
```

33.3.3 1c. Note the secret key and version number

The output of the command above contains important data for Step 2e below:

```
Using legacy password setting method
Failed to set property 'servicePrincipalName' to 'HTTP/<zammad-host>' on Dn 'CN=Zammad_Service,DC=<domain>,DC=<tld>': 0x13.
WARNING: Unable to set SPN mapping data.
If <service-acct> already has an SPN mapping installed for HTTP/<zammad-host>, this is no cause for concern.
Building salt with principalname HTTP/<zammad-host> and domain <domain> (encryption type 18)... Hashing password with salt "<domain><service-acct>".
Key created.
Output keytab to zammad.keytab:
Keytab version: 0x502
keysize 67 <service-acct>@<domain> ptype 1 (KRB5_NT_PRINCIPAL) vno 3 etype 0x12 (AES256-SHA1) keylength 32 (0x5ee827c30c736dd4095c9cbe146eabc216415b1dddb134db6aabd61be8fdf7fb1)
```

On the last line, take note of:

- **the secret key** in parentheses at the end (`0x5ee827...`)
- **the secret key version number** preceded by `vno` (3)
33.4 Step 2: Remove NGINX, Set up Apache + Kerberos

Next, the Zammad host must be configured to support Kerberos (and to accept auth credentials provided by the Active Directory server).

In most cases, you would have to recompile NGINX from source with an extra module to enable Kerberos support. To get around this, we will use Apache, which offers Kerberos support through a plug-in module instead.

**Note:** All commands in this section must be run as root (or with `sudo`).

### 33.4.1 2a. Turn off NGINX

```bash
$ systemctl stop nginx  # turn off nginx
$ systemctl disable nginx  # keep it off after reboot
```

**Warning:** This will take your Zammad instance **offline** until Apache is fully configured and running.

If you wish to minimize downtime, you can save this step for last; just bear in mind that Apache will not start if the port it wants to listen on is being used by NGINX.

If for any reason you can’t complete this tutorial, simply turn off Apache and restore NGINX:

```bash
$ systemctl stop apache2
$ systemctl disable apache2
$ systemctl enable nginx
$ systemctl start nginx
```

### 33.4.2 2b. Pre-Configure Apache

This documentation expects a already working Apache configuration. Please see *Configure the webserver* before continuing.

### 33.4.3 2c. Install further Apache dependencies

**Ubuntu / Debian**

**CentOS**

**OpenSUSE**

```bash
$ apt update
$ apt install krb5-user libapache2-mod-auth-kerb

$ yum install krb5-workstation mod_auth_kerb

$ zypper ref
$ zypper install krb5-client apache2-mod_auth_kerb
```
33.4.4 2d. Enable Apache modules

SSO requires modules that are not enabled by default. By default you can use `a2enmod` to do so.

```
a2enmod auth_kerb rewrite
$ systemctl restart apache2
```

add/uncomment the appropriate `LoadModule` statements in your Apache config:

```
# /etc/httpd/conf/httpd.conf
LoadModule auth_kerb_module /usr/lib/apache2/modules/mod_auth_kerb.so
LoadModule rewrite_module modules/mod_rewrite.so
```

33.4.5 2e. Configure Kerberos

Kerberos realm configuration is how you tell the Zammad server how to reach the domain controller (Active Directory server).

**Note:** Replace the following placeholders in the sample config below:

- `<domain>` Windows domain
- `<domain-controller>` Domain controller IP/FQDN(s)
- `<master-domain-controller>` Master domain controller IP/FQDN
  (must not be read-only, but can be the same as `<domain-controller>`)

```
# /etc/krb5.conf

[libdefaults]
default_realm = <DOMAIN>
default_tkt_enctypes = aes256-cts-hmac-sha1-96
default_tgs_enctypes = aes256-cts-hmac-sha1-96
permitted_enctypes = aes256-cts-hmac-sha1-96
kdc_timesync = 1
ccache_type = 4
forwardable = false
proxiable = false
fcc-mit-ticketflags = false

[realms]
  # multiple KDCs ok (one `kdc = ...` definition per line)
  <DOMAIN> = {
    kdc = <domain-controller>
    admin_server = <master-domain-controller>
    default_domain = <domain>
  }
```
33.4.6 2f. Generate keytab

Apache needs a Kerberos keytab (key table) to manage its shared secrets with the domain controller.

**Note:** Replace the following placeholders in the commands below:

- `<zammad-host>` Zammad FQDN
- `<domain>` Windows domain
- `<secret-key>` Secret key (*omit the leading 0x*)
- `<vno>` Secret key version number

The secret key and version number were found in *Step 1: Configure Active Directory* above.

```bash
$ ktutil
ktutil: addent -key -p HTTP/<zammad-host>@<domain> -k <vno> -e aes256-cts

Key for HTTP/<zammad-host>@<domain> (hex): <secret-key>

ktutil: list
# confirm the entry was added successfully

slot KVNO Principal
----- ---- ---------------------------------------------------------------
 1 3 HTTP/<zammad-host>@<domain>

ktutil: wkt /root/zammad.keytab
# write keytab to disk

ktutil: quit
```

Then, place the keytab in the Apache config directory and set the appropriate permissions:

Ubuntu, Debian, openSUSE

CentOS

```bash
$ mv /root/zammad.keytab /etc/apache2/
$ chown www-data:www-data /etc/apache2/zammad.keytab
$ chmod 400 /etc/apache2/zammad.keytab

$ mv /root/zammad.keytab /etc/httpd/
$ chown apache:apache /etc/httpd/zammad.keytab
$ chmod 400 /etc/httpd/zammad.keytab
```
33.4.7 2g. Configure Apache

Add the following directive to the end of the virtual host configuration file to create your Kerberos SSO endpoint at /auth/sso:

```
<LocationMatch ^/auth/sso$>
  SSLRequireSSL
  AuthType Kerberos
  AuthName "Your Zammad"
  KrbMethodNegotiate On
  KrbVerifyKDC On
  KrbMethodK5Passwd On
  KrbAuthRealms <DOMAIN>
  KrbLocalUserMapping on       # strips @REALM suffix from REMOTE_USER
  KrbServiceName HTTP/<zammad-host>@<DOMAIN>
  Krb5KeyTab /etc/apache2/zammad.keytab # Ubuntu, Debian, & openSUSE
  Krb5KeyTab /etc/httpd/zammad.keytab    # CentOS
  require valid-user

  RequestHeader set X-Forwarded-User "%{RU}e" env=RU
</LocationMatch>
```

Note: Replace the following placeholders in the command below:

- `<zammad-host>` Zammad FQDN
- `<domain>` Windows domain

The configuration below contains two Krb5KeyTab lines! Keep only the one you need.

33.4.8 2g. Restart Apache to apply changes

```
$ systemctl restart apache2
```

33.5 Step 3: Enable SSO in Zammad

Next, enable “Authencation via SSO” in Zammad’s Admin Panel under Settings > Security > Third-Party Applications:

Note: On older versions of Zammad, visit https://your.zammad.host/auth/sso to sign in.
Fig. 3: In Zammad 3.5, this option adds a Sign in using SSO button to the sign-in page.

### 33.6 Step 4: Configure Client System (Windows Only)

For the full SSO experience (i.e., for passwordless, one-click sign-in), Zammad users must:

1. be on the Active Directory server’s local intranet; and
2. modify their network settings for the Zammad host to be treated as a local intranet server.

Fig. 4: Without this step, users must enter their Active Directory credentials during SSO.

IE / Edge / Chromium
Firefox

**Tip:** This setting can be centrally managed across the entire intranet using a group policy object (GPO).
1. Add your Zammad FQDN in Internet Options under **Security > Local Intranet > Sites > Advanced**.
2. Select “Require server verification (https:) for all sites in this zone”.
3. Under **Security level for this zone > Custom level… > Settings > User Authentication > Logon**, select “Automatic logon only in Intranet Zone”.

**Note:** This option cannot be centrally managed because it is set in the browser rather than Windows Settings.

1. Enter `about:config` in the address bar. Click **Accept the risk and continue**.
2. Search for the `network.negotiate-auth.trusted-uris` option.
3. Double-click to edit, then add your Zammad FQDN.
4. Restart Firefox to apply your changes.

![Fig. 5: Enter about:config in the address bar to access advanced settings in Firefox.](image)

### 33.7 Troubleshooting

- Are all relevant FQDNs/hostnames reachable from your Active Directory and Zammad servers (including each other’s)?
- Are the system clocks of your Active Directory and Zammad servers synchronized within five minutes of each other? (Kerberos is a time-sensitive protocol.)

#### 33.7.1 Errors in Apache Logs

**Tip:** Try raising your Apache log level temporarily.

Add `LogLevel debug` to your virtual host configuration, then restart the service to apply the changes.

“*an unsupported mechanism was requested*” Does your Active Directory service account have **Kerberos AES 256-bit encryption** enabled?

If for some reason your server does not support AES 256-bit encryption, the LDAP Wiki has [more information about Kerberos encryption types](https://ldapwiki.com/wiki/Kerberos).

“*failed to verify krb5 credentials: Key version is not available*” Did you use the exact **version number** (vno) provided by `ktpass` when *generating your keytab*?

Try generating it again, just to be sure.

“*unspecified GSS failure. Minor code may provide more information (, No key table entry found for HTTP/FQDN@DOMAIN)*” Does the **service name** you provided to `setspn` exactly match the one you used when *generating your keytab*?

Try generating it again, just to be sure.

“*No key table entry found for HTTP/FQDN@DOMAIN*” Does your virtual host configuration’s **KrbServiceName** setting exactly match the **service name** you provided to `setspn`?

This setting is case-sensitive.
“Warning: received token seems to be NTLM, which isn’t supported by the Kerberos module. Check your IE configuration”

Is your Zammad host accessible at an FQDN? This error may indicate that you configured your Zammad host as a numeric IP address instead.

“Cannot decrypt ticket for HTTP/FQDN@DOMAIN” Did you make sure to change the password on your Active Directory service account after enabling 256-bit AES encryption?

And did you make sure to register the SPN (with ktpass) and generate your keytab (with ktutil) after changing your password?

Try running kinit -k -t <path to keytab> HTTP/<zammad-host>@<DOMAIN>. If no output is returned, you’re good - if you see “kinit: Preauthentication failed while getting initial credentials” your credentials provided were wrong or you used /pass * during ktpass command.

“failed when verifying KDC” and “failed to verify krb5 credentials: Decrypt integrity check failed” Ensure KrbServiceName is the correct ServiceName provided via setspn.

Ensure your Active Directory supports the encryption method configured.

If all above is correct and the rest of FAQ also is ensured, make sure your client does not cache the results. klist purge clears the clients cache - a reboot of your client would do too.
This guide will discuss how to set up third party reporting tools with Zammad.

Fig. 1: Use third party reporting tools to boost your reporting capabilities.

### 34.1 Getting Started

You will need

- A instance of the reporting tool of your choice (hosted or self-hosted)
- (read) access to your Elasticsearch index

**Warning:** Never expose Elasticsearch to the public if you’re not sure how to do it. Especially *never* without authentication! Zammad stores very sensitive information within the Elasticsearch Index.

- (a Zammad 4+ instance that supports your use case)

**Note:** Limitations

Please note that this guide expects all requirements to be up and running. We will not cover core configurations of each tool. Please also note that we can’t support you with configuration of your specific third party tool.
Specific use cases

You may have specific use cases which we can’t cover in this documentation. The following sub pages and also our List of Indexed Attributes should provide enough information to help you!

34.2 Third Party Reporting Tools known to be working

34.2.1 Grafana

Grafana allows you to query, visualize and alert on metrics your Zammad installation stores within the Elasticsearch indexes.

Overview

Quickly jump to…

• Setting up required data sources
• The Dashboards

You will need

• A Grafana 7.5+ instance (hosted or self hosted)
• Worldmap panel plugin
• (read) access to your Elasticsearch index

Warning: Never expose Elasticsearch to the public if you’re not sure how to do it. Especially never without authentication! Zammad stores very sensitive information within the Elasticsearch Index.

• (a Zammad 4+ instance that supports your use case)
Note: Limitations

Please note that this guide expects all requirements to be up and running. We will not cover core configurations of each tool. Please also note that we can’t support you with configuration of your specific third party tool.

Specific use cases

You may have specific use cases which we can’t cover in this documentation. The following sub pages and also our List of Indexed Attributes should provide enough information to help you!

Setting up required data sources

Hint: You may not need all data sources

Before we start: The data sources always follow the same scheme. We reduced below information to name, time field name and index name. Everything else relies on your environment and is out of our scope.

Note: Please replace zammad_production_ with your fitting prefix.

ES - Chat Sessions:
  Index name: zammad_production_chat_session
  Time field name: created_at

ES - CTI Log:
  Index name: zammad_production_cti_log
  Time field name: start_at

ES - Ticket Articles:
  Index name: zammad_production_ticket
  Time field name: article.created_at

ES - Tickets by closed_at:
  Index name: zammad_production_ticket
  Time field name: close_at

ES - Tickets by created_at:
  Index name: zammad_production_ticket
  Time field name: created_at

ES - Tickets by first_response_at:
  Index name: zammad_production_ticket
  Time field name: first_response_at

With above data sources you basically have everything you need to start building your own dashboards.

Tip: Not sure about your index names?
Querying your Elasticsearch like below
# Replace localhost:9200 with the IP/URL of your setup if needed

```bash
$ curl http://localhost:9200/_aliases?pretty=true
```

will return a list that looks similar to the following:

```json
{
    "zammad_production_knowledge_base_translation" : {
        "aliases" : { }
    },
    "zammad_production_ticket_priority" : {
        "aliases" : { }
    },
    "zammad_production_stats_store" : {
        "aliases" : { }
    },
    "zammad_production_organization" : {
        "aliases" : { }
    },
    "zammad_production_cti_log" : {
        "aliases" : { }
    },
    "zammad_production_group" : {
        "aliases" : { }
    },
    "zammad_production_knowledge_base_answer_translation" : {
        "aliases" : { }
    },
    "zammad_production_ticket" : {
        "aliases" : { }
    },
    "zammad_production_ticket_state" : {
        "aliases" : { }
    },
    "zammad_production_chat_session" : {
        "aliases" : { }
    },
    "zammad_production_user" : {
        "aliases" : { }
    },
    "zammad_production_knowledge_base_category_translation" : {
        "aliases" : { }
    }
}
```
The Dashboards

If you want to get inspired, you can also use our sample dashboards as mentioned below. These dashboards can also be found on GitHub.

Importing an existing Dashboard

Navigate to → Import and either upload the json file you received or use the grafana.com ID. During importing you can provide a dashboard name and folder. You’ll also be asked to map the data sources to your environment. If you used our data source names above, you can simply search for the same name.

![Fig. 2: Importing existing dashboards by ID](image)

Ticket statistics

Tip: Grafana.com ID: 14222

![Dashboard screenshot](image)

This dashboard provides graphs for:

- ticket opening and closing
- created articles
- ticket SLA (in time and violation) per type

It also contains specific ticket and article meta information:

- ticket group distribution

---

2 Some values are not available as time series information. This means we can only display the last value of the field in question.

3 Requires SLA function to be active. Negative values indicate SLA violations.
• sender ratio (e.g. Customer / Agent)
• article type ratio (e.g. email, phone)
• article content type
• escalation ratios
• average first response, update time and close time
• top 10
  – organization of ticket customer
  – ticket customers
  – ticket owners
  – average accounted time on ticket
  – ticket tags
• last 10 escalated tickets

Required data sources:
• ES - Ticket Articles
• ES - Tickets by created_at
• ES - Tickets by closed_at

Chat-Session statistics

Tip: Grafana.com ID: 14224

This dashboard provides graphs for:

1 Specific reference IDs are not the same on every instance and thus the panel may not work or show incorrect data. Check the panels description on how to find our the relations on your system.
• Chat session creations

It also contains specific chat session meta information:

• top 10
  – chat tags
  – chat agents
  – chat exit pages
  – city origins
• chat topic ratio
• average number of messages within chat-sessions
• average chatting time
• World map with chat origin countries

Required data sources:

• ES - Chat Sessions

CTI-Log statistics

Tip: Grafana.com ID: 14223

This dashboard provides graphs for:

• number of calls per direction (in / out)

It also contains specific chat session meta information:

• call ratio (in / out)
• average waiting time
• average talking time
• top 10
  – callers (in)
  – call answerers (in)

Required data sources:
• ES – CTI Log

Note: Your favorite tool is not available?
Worry not, if it does support Elasticsearch Indexes, you may be good to go! See *List of Indexed Attributes* for available indexes.