

Configuring GAB-Sync for Kopano OL Extension

- [Configuring the GAB-Sync Script](#)
- [Hidden GAB entries](#)
- [Automating execution of z-push-gabsync](#)
- [Download](#)

The Kopano OL Extension (KOE) serves as an enhancement to the already included ActiveSync support of Microsoft Outlook 2013 and 2016 (Windows only, but for both 32bit and 64bit systems). KOE is available as an msi package and can be installed before or after creating the actual ActiveSync account in Outlook.

The client side documentation has been moved over to the [Kopano Knowledge Base](#).

Configuring the GAB-Sync Script

The scripts and configuration files in the following article are part of the package **z-push-kopano-gabsync**

The GAB-Sync script can be run on any host that has the php-mapi module installed and that has access to your Zarafa/Kopano system. The script also brings its own config file (*config.php*), which needs to be customised to the local environment, so that the script can login at your Kopano/Zarafa system with admin privileges.

Amount of Chunks for GAB data

Depending on the total amount of GAB entries *AMOUNT_OF_CHUNKS* could be adjusted in the gab-sync *config.php* to have smaller chunks of data.

Excerpt of the gab-sync config.php. *SERVER*, *USERNAME* and *PASSWORD* may need to be adjusted to the local installation if the script is running on a different system than your Zarafa/Kopano installation. If Z-Push and Zarafa/Kopano are running on the same system no adjustments are necessary.

Multi-Tenant/Hosted Environments

For multi-tenant systems it's not possible to authenticate using 'SYSTEM'. A user with full system admin privileges is required (*kopanoAdmin=2* in LDAP).

```
// Server connection settings
// Depending on your setup, it might be advisable to change the lines
// below to one defined with your
// default socket location.
// Normally "default:" points to the default setting
("file:///var/run/kopano/server.sock")
// Examples: define("SERVER", "default:");
// define("SERVER", "http://localhost:236/kopano");
// define("SERVER", "https://localhost:237/kopano");
// define("SERVER", "file:///var/run/kopano/server.sock");
// If you are using ZCP >= 7.2.0, set it to the zarafa location, e.g.
// define("SERVER", "http://localhost:236/zarafa");
// define("SERVER", "https://localhost:237/zarafa");
// define("SERVER", "file:///var/run/zarafad/server.sock");
// For ZCP versions prior to 7.2.0 the socket location is different
// (http(s) sockets are the same):
// define("SERVER", "file:///var/run/zarafa");
define('SERVER', 'file:///var/run/zarafad/server.sock');
define('USERNAME', 'SYSTEM');
define('PASSWORD', '');
define('CERTIFICATE', null);
define('CERTIFICATE_PASSWORD', null);
```

Runtime example of *gab-sync.php*:

```

root@webapp:/usr/share/z-push/tools/gab-sync# ./gab-sync.php -a sync
Kopano->openMessageStore(): Found 'PUBLIC' store of user 'SYSTEM':
'Resource id #15'
Starting GAB sync to store 'SYSTEM' on id
'97b7db77f54d4666ab345a2b8b13f26bed3000000000'
Kopano->ClearAllNotCurrentChunkType: no invalid items, done!
Kopano->SetChunkData: account-10/7      Entries: 6      Size: 4192 B
CRC: a3ecb0b6a47a84896c3a702d5e13992a - unchanged
Kopano->SetChunkData: account-10/0      Entries: 5      Size: 3416 B
CRC: 70306c3e9c5ce790adf76c4ffdb2c955 - unchanged
Kopano->SetChunkData: account-10/3      Entries: 5      Size: 3200 B
CRC: 1f6ce75c524741ee4385e15d49b66634 - unchanged
Kopano->SetChunkData: account-10/2      Entries: 7      Size: 5607 B
CRC: 0352020d458ef1f57fd1e7bbff79112f - unchanged
Kopano->SetChunkData: account-10/5      Entries: 10     Size: 6538 B
CRC: 22ca82af353d424cb49a80a3f78b4847 - unchanged
Kopano->SetChunkData: account-10/9      Entries: 10     Size: 6480 B
CRC: d3ee2b86fc6764a9884d57893892379f - unchanged
Kopano->SetChunkData: account-10/1      Entries: 3      Size: 1902 B
CRC: a3d17f8ac0e0d0379d5fc45e3df01206 - unchanged
Kopano->SetChunkData: account-10/6      Entries: 8      Size: 5265 B
CRC: 360ce84f0937f98a28d66b3ecf84c8e2 - unchanged
Kopano->SetChunkData: account-10/4      Entries: 4      Size: 2566 B
CRC: f85f6242fe06a6b969c47aaab30537b5 - unchanged
Kopano->SetChunkData: account-10/8      Entries: 4      Size: 2575 B
CRC: 59b3a3c1692892bd5b686b22d982a7df - unchanged
Sync:
    Items in GAB:                62
    Total data size:              41741 B
    Avg. of items per chunk:      6.2
    Min. of items per chunk:      3
    Max. of items per chunk:      10
    Avg. of size per chunk:       4174 B
    Min. of size per chunk:       1902 B
    Max. of size per chunk:       6538 B
    Configured amount of chunks:  10
    Ideal amount by entries:       10
    Ideal amount by size:         5

```

The id given in "Starting GAB sync to store 'SYSTEM' on id '97b7db77f54d4666ab345a2b8b13f26bed3000000000'" can be inserted into the `KOE_GAB_FOLDERID` variable of Z-Push's configuration file. If left empty Z-Push will search the folder by name and automatically discover the id.

```
// To synchronize the GAB KOE, the GAB store and folderid need to be
specified.
// Use the gab-sync script to generate this data. The name needs to
// match the config of the gab-sync script.
// More information here: https://wiki.z-hub.io/x/z4Aa (GAB Sync Script)
define('KOE_GAB_STORE', 'SYSTEM');
define('KOE_GAB_FOLDERID',
'97b7db77f54d4666ab345a2b8b13f26bed3000000000');
define('KOE_GAB_NAME', 'Z-Push-KOE-GAB');
```

Store of the user SYSTEM

The store associated with the user *SYSTEM* is the public store, available to all users by default. If your system does not yet have a public store, you need to run *kopano-admin -s*.

Make sure enough permissions are set on the public store so users can see the *Z-Push-KOE-GAB* folder. At least "read-only" and "folder visible" permissions are required to get the GAB sync to KOE.

Multi-Tenant/Hosted Environments

The *gab-sync* script itself automatically detects if multiple tenants are configured in an environment and exports the GAB to the public folder of each tenant. For Multi-Tenant Environments the *KOE_GAB_FOLDERID* always needs to be kept empty.

Hidden GAB entries

If contacts or accounts are marked as hidden in the GAB they were still visible in Outlook when using this script. This is done via the *kopanoHidden* or *zarafaHidden* flag (depending on your LDAP schema).

With Z-Push 2.3.4 this was fixed, but versions prior to KC 8.2 require additional configuration.

It's required to add the following line to your */etc/kopano/ldap.propmap.cfg*

```
# PR_EC_AB_HIDDEN
0x67A7000B = kopanoHidden
```

If you use the Zarafa LDAP schema you need do the same, but changing */etc/zarafa/ldap.propmap.cfg* adding the mapping of the '*zarafaHidden*' flag.

When using KC 8.2 or newer, check */usr/share/kopano/ldap.propmap.cfg* if this already mapped.

After doing this change it's required to restart *kopano-server*.

Executing the *sync-gab* script will output an additional line for hidden users, like:

```
Kopano->GetGAB(): Ignoring user 'info-redirect' as account is hidden
```

Automating execution of z-push-gabsync

Depending on the amount of changes to the GAB of your Kopano system you should setup a cronjob to invoke *z-push-gabsync* from time to time. For normal system we recommend to add the following snippet to your daily cron run (e.g. */etc/cron.daily*):

```
#!/bin/sh
set -e

GABSYNC=/usr/share/z-push/tools/gab-sync/gab-sync.php

if [ -e $GABSYNC ]; then
    $GABSYNC -a sync
fi
```

If Z-Push is running on the same system as your Kopano, then you could also add the above script to `/etc/kopano/userscripts/createuser.d` to additionally run it whenever a new user has been created.

Download

The Kopano OL Extension is available for Kopano partners and customers from <https://download.kopano.io/supported/olextension/> (Login with credentials from the Kopano Partner Portal).