GNU Taler Exchange Manual

Release 0.6.0pre1

GNU Taler team

Sep 19, 2019

CONTENTS

CHAPTER

ONE

INTRODUCTION

This manual is an early draft that still needs significant editing work to become readable.

1.1 About GNU Taler

GNU Taler is an open protocol for an electronic payment system with a free software reference implementation. GNU Taler offers secure, fast and easy payment processing using well understood cryptographic techniques. GNU Taler allows customers to remain anonymous, while ensuring that merchants can be held accountable by governments. Hence, GNU Taler is compatible with anti-money-laundering (AML) and know-your-customer (KYC) regulation, as well as data protection regulation (such as GDPR).

GNU Taler is not yet production-ready, after following this manual you will have a backend that can process payments in "KUDOS", but not regular currencies. This is not so much because of limitations in the backend, but because we are not aware of a Taler exchange operator offering regular currencies today.

1.2 About this manual

This tutorial targets system administrators who want to install and operate a GNU Taler exchange.

1.3 Organizational prerequisites

Operating a GNU Taler exchange means that you are operating a payment service provider, which means that you will most likely need a bank license and/or follow applicable financial regulation.

GNU Taler payment service providers generally need to ensure high availability and have *really* good backups (synchronous replication, asynchronous remote replication, off-site backup, 24/7 monitoring, etc.).[1] This manual will not cover these aspects of operating a payment service provider.

We will assume that you can operate a (high-availability, high-assurance) Postgres database. Furthermore, we expect some moderate familiarity with the compilation and installation of free software packages. You need to understand the cryptographic concepts of private and public keys and must be able to protect private keys stored in files on disk. An exchange uses an *offline* master key as well as *online* keys. You are advised to secure your private master key and any copies on encrypted, always-offline computers. Again, we assume that you are familiar with good best practices in operational security, including securing key material.[2]

[1] Naturally, you could operate a Taler exchange for a toy currency without any real value on low-cost setups like a Raspberry Pi, but we urge you to limit the use of such setups to research and education as with GNU Taler data loss instantly results in financial losses.

[2] The current implementation does not make provisions for secret splitting. Still, the use of a hardware security module (HSM) for protecting private keys is adviseable, so please contact the developers for HSM integration support.

1.4 Architecture overview

Taler is a pure payment system, not a new crypto-currency. As such, it operates in a traditional banking context. In particular, this means that in order to receive funds via Taler, the merchant must have a regular bank account, and payments can be executed in ordinary currencies such as USD or EUR. Similarly, the Taler exchange must interact with a bank. The bank of the exchange holds the exchange's funds in an escrow account.

When customers wire money to the escrow account, the bank notifies the exchange about the incoming wire transfers. The exchange then creates a *reserve* based on the subject of the wire transfer. The wallet which knows the secret key matching the wire transfer subject can then withdraw coins from the reserve, thereby draining it. The liability of the exchange against the reserve is thereby converted into a liability against digital coins issued by the exchange. When the customer later spends the coins at a merchant, and the merchant *deposits* the coins at the exchange, the exchange first *aggregates* the amount from multiple deposits from the same merchant and then instructs its bank to make a wire transfer to the merchant, thereby fulfilling its obligation and eliminating the liability. The exchange charges *fees* for some or all of its operations to cover costs and possibly make a profit.

Auditors are third parties, for example financial regulators, that verify that the exchange operates correctly. The same software is also used to calculate the exchange's profits, risk and liabilities by the accountants of the exchange.

The Taler software stack for an exchange consists of the following components:

- HTTP frontend The HTTP frontend interacts with Taler wallets and merchant backends. It is used to withdraw coins, deposit coins, refresh coins, issue refunds, map wire transfers to Taler transactions, inquire about the exchange's bank account details, signing keys and fee structure. The binary is the `taler-exchange-httpd`.
- Aggregator The aggregator combines multiple deposits made by the same merchant and (eventually) triggers wire transfers for the aggregate amount. The merchant can control how quickly wire transfers are made. The exchange may be charge a fee per wire transfer to discourage excessively frequent transfers. The binary is the `taler-exchange-aggregator`.
- Auditor The auditor verifies that the transactions performed by the exchange were done properly. It checks the various signatures, totals up the amounts and alerts the operator to any inconsistencies. It also computes the expected bank balance, revenue and risk exposure of the exchange operator. The main binary is the `taler-auditor`.
- Wire plugin A wire plugin enables the HTTP frontend to talk to the bank. Its role is to allow the exchange to validate bank addresses (i.e. IBAN numbers), for the aggregator to execute wire transfers and for the auditor to query bank transaction histories. Wire plugins are *plugins* as there can be many different implementations to deal with different banking standards. Wire plugins are automatically located and used by the exchange, aggregator and auditor.
- DBMS Postgres The exchange requires a DBMS to stores the transaction history for the Taler exchange and aggregator, and a (typically separate) DBMS for the Taler auditor. For now, the GNU Taler reference implemenation only supports Postgres, but the code could be easily extended to support another DBMS.

CHAPTER

TWO

INSTALLATION

Please install the following packages before proceeding with the exchange compilation.

- GNU autoconf >= 2.69
- GNU automake >= 1.14
- GNU libtool >= 2.4
- GNU autopoint >= 0.19
- GNU libltdl >= 2.4
- GNU libunistring >= 0.9.3
- libcurl >= 7.26 (or libgnurl >= 7.26)
- GNU libmicrohttpd >= 0.9.59
- GNU libgcrypt >= 1.6
- libjansson >= 2.7
- Postgres >= 9.6, including libpq
- libgnunetutil (from Git)
- GNU Taler exchange (from Git)

Except for the last two, these are available in most GNU/Linux distributions and should just be installed using the respective package manager.

The following instructions will show how to install libgnunetutil and the GNU Taler exchange.

Before you install libgnunetutil, you must download and install the dependencies mentioned above, otherwise the build may succeed but fail to export some of the tooling required by Taler.

To download and install libgnunetutil, proceed as follows:

```
$ git clone https://git.gnunet.org/gnunet/
$ cd gnunet/
$ ./bootstrap
$ ./configure [--prefix=GNUNETPFX]
$ # Each dependency can be fetched from non standard locations via
$ # the '--with-<LIBNAME>' option. See './configure --help'.
$ make
# make install
```

If you did not specify a prefix, GNUnet will install to `/usr/local`, which requires you to run the last step as `root`.

To download and install the GNU Taler exchange, proceeds as follows:

```
$ git clone git://git.taler.net/exchange
$ cd exchange
$ ./bootstrap
$ ./configure [--prefix=EXCHANGEPFX] \
              [--with-gnunet=GNUNETPFX]
$ # Each dependency can be fetched from non standard locations via
$ # the '--with-<LIBNAME>' option. See './configure --help'.
$ make
# make install
```

If you did not specify a prefix, the exchange will install to `/usr/local`, which requires you to run the last step as `root`. Note that you have to specify `--with-gnunet=/usr/local` if you installed GNUnet to `/usr/local` in the previous step.

CHAPTER

THREE

CONFIGURATION

This chapter provides an overview of the exchange configuration. Or at least eventually will do so, for now it is a somewhat wild description of some of the options.

3.1 Configuration format

configuration In Taler realm, any component obeys to the same pattern to get configuration values. According to this pattern, once the component has been installed, the installation deploys default values in ${prefix}/share/taler/config.d/, in .conf files. In order to override these defaults, the user can write a custom .conf file and either pass it to the component at execution time, or name it taler.conf and place it under $HOME/.config/.

A config file is a text file containing sections, and each section contains its values. The right format follows:

```
[section1]
value1 = string
value2 = 23

[section2]
value21 = string
value22 = /path22
```

Throughout any configuration file, it is possible to use `$`-prefixed variables, like `$VAR`, especially when they represent filesystem paths. It is also possible to provide defaults values for those variables that are unset, by using the following syntax: `${VAR:-default}`. However, there are two ways a user can set `$`-prefixable variables:

by defining them under a `[paths]` section, see example below,

```
[paths]
TALER_DEPLOYMENT_SHARED = ${HOME}/shared-data
..
[section-x]
path-x = ${TALER_DEPLOYMENT_SHARED}/x
```

or by setting them in the environment:

```
$ export VAR=/x
```

The configuration loader will give precedence to variables set under `[path]`, though.

The utility `taler-config`, which gets installed along with the exchange, serves to get and set configuration values without directly editing the .conf. The option `-f` is particularly useful to resolve pathnames, when they use several levels of `$`-expanded variables. See `taler-config --help`.

Note that, in this stage of development, the file `$HOME/.config/taler.conf` can contain sections for *all* the component. For example, both an exchange and a bank can read values from it.

The repository `git://taler.net/deployment` contains examples of configuration file used in our demos. See under `deployment/config`.

> **Note**
>
> Expectably, some components will not work just by using default values, as their work is often interdependent. For example, a merchant needs to know an exchange URL, or a database name.

3.2 Using taler-config

The tool `taler-config` can be used to extract or manipulate configuration values; however, the configuration use the well-known INI file format and can also be edited by hand.

Run

```
$ taler-config -s $SECTION
```

to list all of the configuration values in section `$SECTION`.

Run

```
$ taler-config -s $section -o $option
```

to extract the respective configuration value for option `$option` in section `$section`.

Finally, to change a setting, run

```
$ taler-config -s $section -o $option -V $value
```

to set the respective configuration value to `$value`. Note that you have to manually restart the Taler backend after you change the configuration to make the new configuration go into effect.

Some default options will use $-variables, such as `$DATADIR` within their value. To expand the `$DATADIR` or other $-variables in the configuration, pass the `-f` option to `taler-config`. For example, compare:

```
$ taler-config -s ACCOUNT-bank \
               -o WIRE_RESPONSE
$ taler-config -f -s ACCOUNT-bank \
               -o WIRE_RESPONSE
```

While the configuration file is typically located at `$HOME/.config/taler.conf`, an alternative location can be specified to `taler-merchant-httpd` and `taler-config` using the `-c` option.

3.3 Keying

The exchange works with three types of keys:

- master key
- sign keys
- denomination keys (see section Coins)
- MASTER_PRIV_FILE: Path to the exchange's master private file.

- MASTER_PUBLIC_KEY: Must specify the exchange's master public key.

3.4 Serving

The exchange can serve HTTP over both TCP and UNIX domain socket.

The following values are to be configured in the section [exchange]:

- serve: must be set to tcp to serve HTTP over TCP, or unix to serve HTTP over a UNIX domain socket
- port: Set to the TCP port to listen on if serve Is tcp.
- unixpath: set to the UNIX domain socket path to listen on if serve Is unix
- unixpath_mode: number giving the mode with the access permission MASK for the unixpath (i.e. 660 = rw-rw—-).

3.5 Currency

The exchange supports only one currency. This data is set under the respective option currency in section [taler].

3.6 Bank account

To configure a bank account in Taler, we need to furnish four pieces of information:

- The `payto://` URL of the bank account, which uniquely idenfies the account. Examples for such URLs include `payto://sepa/CH9300762011623852957` for a bank account in the single European payment area (SEPA) or `payto://x-taler-bank/localhost:8080/2` for the 2nd bank account a the Taler bank demonstrator running at `localhost` on port 8080. The first part of the URL following `payto://` ("sepa" or "x-taler-bank") is called the wire method.
- A matching wire plugin that implements a protocol to interact with the banking system. For example, the EBICS plugin can be used for SEPA transfers, or the "taler-bank" plugin can interact with the Taler bank demonstrator. A wire plugin only supports one particular wire method. Thus, you must make sure to pick a plugin that supports the wire method used in the URL.
- A file containing the signed JSON-encoded bank account details for the /wire API. This is necessary as Taler supports offline signing for bank accounts for additional security.
- Finally, the plugin needs to be provided resources for authentication to the respective banking service. The format in which the authentication information must be provided depends on the wire plugin.

You can configure multiple accounts for an exchange by creating sections starting with "account-" for the section name. You can ENABLE for each account whether it should be used, and for what (incoming or outgoing wire transfers):

```
[account-1]
URL = "payto://sepa/CH9300762011623852957"
WIRE_RESPONSE = ${TALER_CONFIG_HOME}/account-1.json

# Currently, only the 'taler_bank' plugin is implemented.
PLUGIN = <plugin_name_here>

# Use for exchange-aggregator (outgoing transfers)
ENABLE_DEBIT = YES
```

(continues on next page)

(continued from previous page)

```
# Use for exchange-wirewatch (and listed in /wire)
ENABLE_CREDIT = YES

# Authentication options for the chosen plugin go here.
# (Next sections have examples of authentication mechanisms)
```

The command line tool taler-exchange-wire is used to create the `account-1.json` file. For example, the utility may be invoked as follows to create all of the WIRE_RESPONSE files (in the locations specified by the configuration):

```
$ taler-exchange-wire
```

The generated file will be echoed by the exchange when serving /wire[3] requests.

3.6.1 Wire plugin "taler_bank"

x-taler-bank taler_bank plugin The `taler_bank` plugin implements the wire method "x-taler-bank".

The format of the `payto://` URL is `payto://x-taler-bank/HOSTNAME[:PORT]`.

For basic authentication, the `taler_bank` plugin only supports simple password-based authentication. For this, the configuration must contain the "USERNAME" and "PASSWORD" of the respective account at the bank.

```
[account-1]

# Bank account details here..
# ..

# Authentication options for the taler_bank plugin below:

TALER_BANK_AUTH_METHOD = basic
USERNAME = exchange
PASSWORD = super-secure
```

3.6.2 Wire plugin "ebics"

The "ebics" wire plugin is not fully implemented and today does not support actual wire transfers.

> **Note**
>
> The rationale behind having multiple bank accounts is that the exchange operator, as a security measure, may want to instruct the bank that the incoming bank account is only supposed to *receive* money.

3.6.3 Wire fee structure

wire fee fee For each wire method ("sepa" or "x-taler-wire", but not per plugin!) the exchange configuration must specify applicable wire fees. This is done in configuration sections of the format `fees-METHOD`. There are two types of fees, simple wire fees and closing fees. Wire fees apply whenever the aggregator transfers funds to a merchant. Closing fees apply whenever the exchange closes a reserve (sending back funds to the customer). The fees must be constant for a full year, which is specified as part of the name of the option.

[3] https://api.taler.net/api-exchange.html#wire-req

```
[fees-iban]
WIRE-FEE-2018 = EUR:0.01
WIRE-FEE-2019 = EUR:0.01
CLOSING-FEE-2018 = EUR:0.01
CLOSING-FEE-2019 = EUR:0.01

[fees-x-taler-bank]
WIRE-FEE-2018 = KUDOS:0.01
WIRE-FEE-2019 = KUDOS:0.01
CLOSING-FEE-2018 = KUDOS:0.01
CLOSING-FEE-2019 = KUDOS:0.01
```

3.7 Database

The option db under section [exchange] gets the DB backend's name the exchange is going to use. So far, only db = postgres is supported. After choosing the backend, it is mandatory to supply the connection string (namely, the database name). This is possible in two ways:

- via an environment variable: TALER_EXCHANGEDB_POSTGRES_CONFIG.
- via configuration option CONFIG, under section [exchangedb-BACKEND]. For example, the demo exchange is configured as follows:

```
[exchange]
...
DB = postgres
...

[exchangedb-postgres]
CONFIG = postgres:///talerdemo
```

3.8 Coins (denomination keys)

Sections specifying denomination (coin) information start with `coin_`. By convention, the name continues with "$CURRENCY_[$SUBUNIT]_$VALUE", i.e. `[coin_eur_ct_10]` for a 10 cent piece. However, only the `coin_` prefix is mandatory. Each `coin_`-section must then have the following options:

- value: How much is the coin worth, the format is CURRENCY:VALUE.FRACTION. For example, a 10 cent piece is "EUR:0.10".
- duration_withdraw: How long can a coin of this type be withdrawn? This limits the losses incurred by the exchange when a denomination key is compromised.
- duration_overlap: What is the overlap of the withdrawal timespan for this coin type?
- duration_spend: How long is a coin of the given type valid? Smaller values result in lower storage costs for the exchange.
- fee_withdraw: What does it cost to withdraw this coin? Specified using the same format as value.
- fee_deposit: What does it cost to deposit this coin? Specified using the same format as value.
- fee_refresh: What does it cost to refresh this coin? Specified using the same format as value.
- rsa_keysize: How many bits should the RSA modulus (product of the two primes) have for this type of coin.

3.9 Keys duration

Both signkeys and denom keys have a starting date. The option lookahead_provide, under section [exchange], is such that only keys whose starting date is younger than lookahead_provide will be issued by the exchange.

signkeys. The option lookahead_sign is such that, being t the time when taler-exchange-keyup is run, taler-exchange-keyup will generate n signkeys, where t + (n * signkey_duration) = t + lookahead_sign. In other words, we generate a number of keys which is sufficient to cover a period of lookahead_sign. As for the starting date, the first generated key will get a starting time of t, and the j-th key will get a starting time of x + signkey_duration, where x is the starting time of the (j-1)-th key.

denom keys. The option lookahead_sign is such that, being t the time when taler-exchange-keyup is run, taler-exchange-keyup will generate n denom keys for each denomination, where t + (n * duration_withdraw) = t + lookahead_sign. In other words, for each denomination, we generate a number of keys which is sufficient to cover a period of lookahead_sign. As for the starting date, the first generated key will get a starting time of t, and the j-th key will get a starting time of x + duration_withdraw, where x is the starting time of the (j-1)-th key.

To change these settings, edit the following values in section [exchange]:

- SIGNKEY_DURATION: How long should one signing key be used?
- LOOKAHEAD_SIGN: How much time we want to cover with our signing keys? Note that if SIGNKEY_DURATION is bigger than LOOKAHEAD_SIGN, `taler-exchange-keyup` will generate a quantity of signing keys which is sufficient to cover all the gap.

CHAPTER

FOUR

DEPLOYMENT

4.1 Keys generation

Once the configuration is properly set up, all the keys can be generated by the tool `taler-exchange-keyup`. The following command generates denomkeys and signkeys, plus the "blob" that is to be signed by the auditor.

```
taler-exchange-keyup -o blob
```

blob contains data about denomkeys that the exchange operator needs to get signed by every auditor he wishes (or is forced to) work with.

In a normal scenario, an auditor must have some way of receiving the blob to sign (Website, manual delivery, ..). Nonetheless, the exchange admin can fake an auditor signature — for testing purposes — by running the following command

```
taler-auditor-sign -m EXCHANGE_MASTER_PUB -r BLOB -u AUDITOR_URL -o OUTPUT_FILE
```

Those arguments are all mandatory.

- `EXCHANGE_MASTER_PUB` the base32 Crockford-encoded exchange's master public key. Tipically, this value lies in the configuration option `[exchange]/master_public_key`.
- `BLOB` the blob generated in the previous step.
- `AUDITOR_URL` the URL that identifies the auditor.
- `OUTPUT_FILE` where on the disk the signed blob is to be saved.

`OUTPUT_FILE` must then be copied into the directory specified by the option `AUDITOR_BASE_DIR` under the section `[exchangedb]`. Assuming `AUDITOR_BASE_DIR = ${HOME}/.local/share/taler/auditors`, the following command will "add" the auditor identified by `AUDITOR_URL` to the exchange.

```
cp OUTPUT_FILE ${HOME}/.local/share/taler/auditors
```

If the auditor has been correctly added, the exchange's `/keys` response must contain an entry in the `auditors` array mentioning the auditor's URL.

4.2 Database upgrades

Currently, there is no way to upgrade the database between Taler versions.

The exchange database can be re-initialized using:

```
$ taler-exchange-dbinit -r
```

However, running this command will result in all data in the database being lost, which may result in significant financial liabilities as the exchange can then not detect double-spending. Hence this operation must not be performed in a production system.

CHAPTER

FIVE

DIAGNOSTICS

This chapter includes various (very unpolished) sections on specific topics that might be helpful to understand how the exchange operates, which files should be backed up. The information may also be helpful for diagnostics.

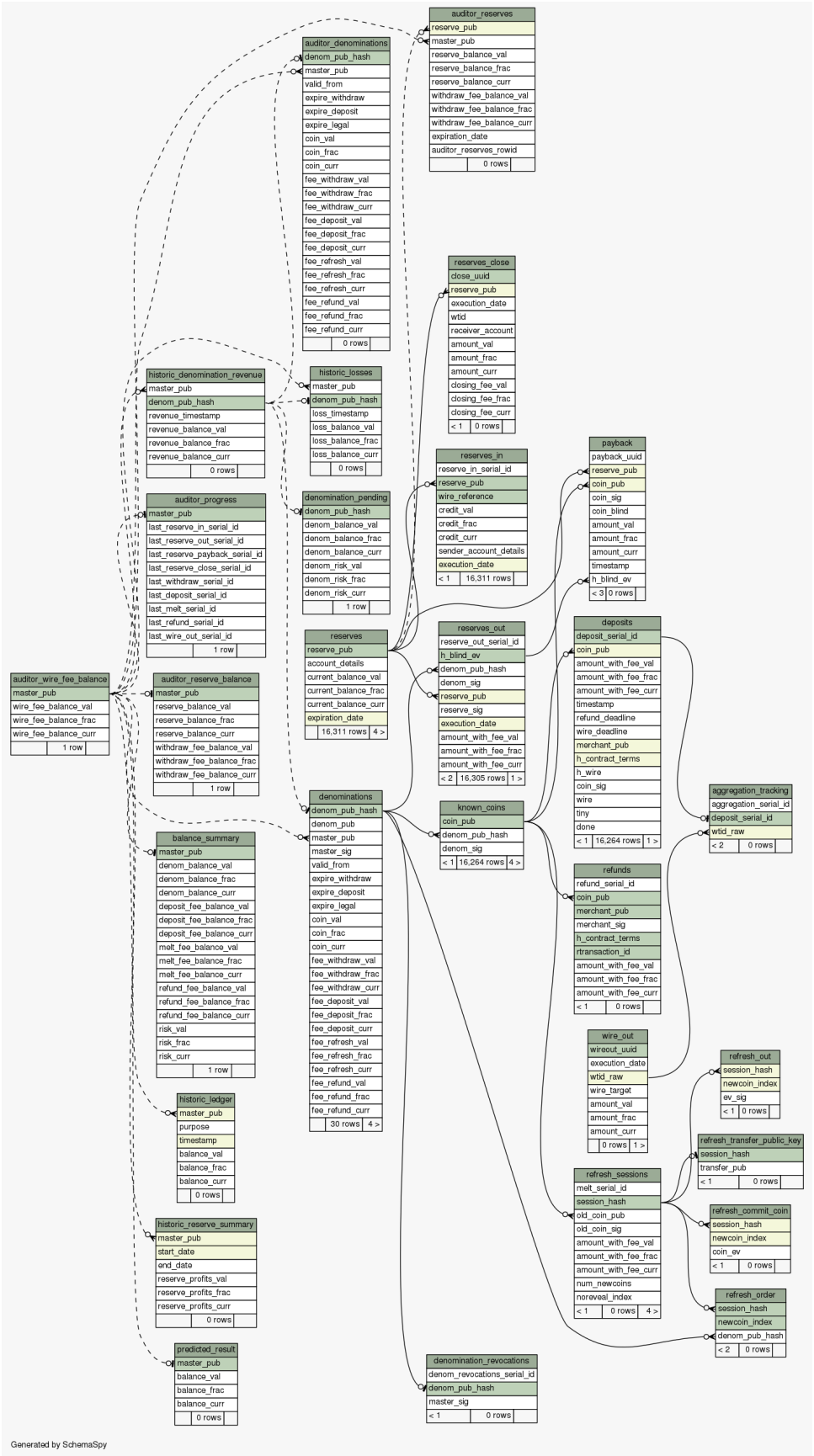
5.1 Reserve management

Incoming transactions to the exchange's provider result in the creation or update of reserves, identified by their reserve key. The command line tool taler-exchange-reservemod allows create and add money to reserves in the exchange's database.

5.2 Database Scheme

The exchange database must be initialized using taler-exchange-dbinit. This tool creates the tables required by the Taler exchange to operate. The tool also allows you to reset the Taler exchange database, which is useful for test cases but should never be used in production. Finally, taler-exchange-dbinit has a function to garbage collect a database, allowing administrators to purge records that are no longer required.

The database scheme used by the exchange look as follows:


auditor_reserves
reserve_pub
master_pub
reserve_balance_val
reserve_balance_frac
reserve_balance_curr
withdraw_fee_balance_val
withdraw_fee_balance_frac
withdraw_fee_balance_curr
expiration_date
auditor_reserves_rowid
0 rows
auditor_denominations
denom_pub_hash
master_pub
valid_from
expire_withdraw
expire_deposit
expire_legal
coin_val
coin_frac
coin_curr
fee_withdraw_val
fee_withdraw_frac
fee_withdraw_curr
fee_deposit_val
fee_deposit_frac
fee_deposit_curr
fee_refresh_val
fee_refresh_frac
fee_refresh_curr
fee_refund_val
fee_refund_frac
fee_refund_curr
0 rows
reserves_close
close_uuid
reserve_pub
execution_date
wtid
receiver_account
amount_val
amount_frac
amount_curr
closing_fee_val
closing_fee_frac
closing_fee_curr
< 1
0 rows
historic_denomination_revenue
master_pub
denom_pub_hash
revenue_timestamp
revenue_balance_val
revenue_balance_frac
revenue_balance_curr
0 rows
historic_losses
master_pub
denom_pub_hash
loss_timestamp
loss_balance_val
loss_balance_frac
loss_balance_curr
0 rows
reserves_in
reserve_in_serial_id
reserve_pub
wire_reference
credit_val
credit_frac
credit_curr
sender_account_details
execution_date
< 1
16,311 rows
payback
payback_uuid
reserve_pub
coin_pub
coin_sig
coin_blind
amount_val
amount_frac
amount_curr
timestamp
h_blind_ev
< 3
0 rows
auditor_progress
master_pub
last_reserve_in_serial_id
last_reserve_out_serial_id
last_reserve_payback_serial_id
last_reserve_close_serial_id
last_withdraw_serial_id
last_deposit_serial_id
last_melt_serial_id
last_refund_serial_id
last_wire_out_serial_id
1 row
denomination_pending
denom_pub_hash
denom_balance_val
denom_balance_frac
denom_balance_curr
denom_risk_val
denom_risk_frac
denom_risk_curr
1 row
reserves
reserve_pub
account_details
current_balance_val
current_balance_frac
current_balance_curr
expiration_date
16,311 rows
4 >
reserves_out
reserve_out_serial_id
h_blind_ev
denom_pub_hash
denom_sig
reserve_pub
reserve_sig
execution_date
amount_with_fee_val
amount_with_fee_frac
amount_with_fee_curr
< 2
16,305 rows
1 >
deposits
deposit_serial_id
coin_pub
amount_with_fee_val
amount_with_fee_frac
amount_with_fee_curr
timestamp
refund_deadline
wire_deadline
merchant_pub
h_contract_terms
h_wire
coin_sig
wire
tiny
done
< 1
16,264 rows
1 >
auditor_wire_fee_balance
master_pub
wire_fee_balance_val
wire_fee_balance_frac
wire_fee_balance_curr
1 row
auditor_reserve_balance
master_pub
reserve_balance_val
reserve_balance_frac
reserve_balance_curr
withdraw_fee_balance_val
withdraw_fee_balance_frac
withdraw_fee_balance_curr
1 row
denominations
denom_pub_hash
denom_pub
master_pub
master_sig
valid_from
expire_withdraw
expire_deposit
expire_legal
coin_val
coin_frac
coin_curr
fee_withdraw_val
fee_withdraw_frac
fee_withdraw_curr
fee_deposit_val
fee_deposit_frac
fee_deposit_curr
fee_refresh_val
fee_refresh_frac
fee_refresh_curr
fee_refund_val
fee_refund_frac
fee_refund_curr
30 rows
4 >
known_coins
coin_pub
denom_pub_hash
denom_sig
< 1
16,264 rows
4 >
aggregation_tracking
aggregation_serial_id
deposit_serial_id
wtid_raw
< 2
0 rows
refunds
refund_serial_id
coin_pub
merchant_pub
merchant_sig
h_contract_terms
rtransaction_id
amount_with_fee_val
amount_with_fee_frac
amount_with_fee_curr
< 1
0 rows
balance_summary
master_pub
denom_balance_val
denom_balance_frac
denom_balance_curr
deposit_fee_balance_val
deposit_fee_balance_frac
deposit_fee_balance_curr
melt_fee_balance_val
melt_fee_balance_frac
melt_fee_balance_curr
refund_fee_balance_val
refund_fee_balance_frac
refund_fee_balance_curr
risk_val
risk_frac
risk_curr
1 row
wire_out
wireout_uuid
execution_date
wtid_raw
wire_target
amount_val
amount_frac
amount_curr
0 rows
1 >
refresh_out
session_hash
newcoin_index
ev_sig
< 1
0 rows
historic_ledger
master_pub
purpose
timestamp
balance_val
balance_frac
balance_curr
0 rows
refresh_transfer_public_key
session_hash
transfer_pub
< 1
0 rows
refresh_sessions
melt_serial_id
session_hash
old_coin_pub
old_coin_sig
amount_with_fee_val
amount_with_fee_frac
amount_with_fee_curr
num_newcoins
noreveal_index
< 1
0 rows
4 >
refresh_commit_coin
session_hash
newcoin_index
coin_ev
< 1
0 rows
historic_reserve_summary
master_pub
start_date
end_date
reserve_profits_val
reserve_profits_frac
reserve_profits_curr
0 rows
refresh_order
session_hash
newcoin_index
denom_pub_hash
< 2
0 rows
predicted_result
master_pub
balance_val
balance_frac
balance_curr
0 rows
denomination_revocations
denom_revocations_serial_id
denom_pub_hash
master_sig
< 1
0 rows
Generated by SchemaSpy

5.3 Signing key storage

The private online signing keys of the exchange are stored in a subdirectory "signkeys/" of the "KEYDIR" which is an option in the "[exchange]" section of the configuration file. The filename is the starting time at which the signing key can be used in microseconds since the Epoch. The file format is defined by the struct TALER_EXCHANGEDB_PrivateSigningKeyInformationP:

```
struct TALER_EXCHANGEDB_PrivateSigningKeyInformationP {
   struct TALER_ExchangePrivateKeyP signkey_priv;
   struct TALER_ExchangeSigningKeyValidityPS issue;
};
```

5.4 Denomination key storage

The private denomination keys of the exchange are store in a subdirectory "denomkeys/" of the "KEYDIR" which is an option in the "[exchange]" section of the configuration file. "denomkeys/" contains further subdirectories, one per denomination. The specific name of the subdirectory under "denomkeys/" is ignored by the exchange. However, the name is important for the "taler-exchange-keyup" tool that generates the keys. The tool combines a human-readable encoding of the denomination (i.e. for EUR:1.50 the prefix would be "EUR_1_5-", or for EUR:0.01 the name would be "EUR_0_01-") with a postfix that is a truncated Crockford32 encoded hash of the various attributes of the denomination key (relative validity periods, fee structure and key size). Thus, if any attributes of a coin change, the name of the subdirectory will also change, even if the denomination remains the same.

Within this subdirectory, each file represents a particular denomination key. The filename is the starting time at which the signing key can be used in microseconds since the Epoch. The format on disk begins with a struct TALER_EXCHANGEDB_DenominationKeyInformationP giving the attributes of the denomination key and the associated signature with the exchange's long-term offline key:

```
struct TALER_EXCHANGEDB_DenominationKeyInformationP {
  struct TALER_MasterSignatureP signature;
  struct TALER_DenominationKeyValidityPS properties;
};
```

This is then followed by the variable-size RSA private key in libgcrypt's S-expression format, which can be decoded using GNUNET_CRYPTO_rsa_private_key_decode().

5.4.1 Revocations

When an exchange goes out of business or detects that the private key of a denomination key pair has been compromised, it may revoke some or all of its denomination keys. At this point, the hashes of the revoked keys must be returned as part of the `/keys` response under "payback". Wallets detect this, and then return unspent coins of the respective denomination key using the `/payback` API.

When a denomination key is revoked, a revocation file is placed into the respective subdirectory of "denomkeys/". The file has the same prefix as the file that stores the struct TALER_EXCHANGEDB_DenominationKeyInformationP information, but is followed by the ".rev" suffix. It contains a 64-byte EdDSA signature made with the master key of the exchange with purpose `TALER_SIGNATURE_MASTER_DENOMINATION_KEY_REVOKED`. If such a file is present, the exchange must check the signature and if it is valid treat the respective denomination key as revoked.

Revocation files can be generated using the `taler-exchange-keyup` command-line tool using the `-r` option. The Taler auditor will instruct operators to generate revocations if it detects a key compromise (which is possible more coins of a particular denomination were deposited than issued).

It should be noted that denomination key revocations should only happen under highly unusual (“emergency”) conditions and not under normal conditions.

5.5 Auditor signature storage

Signatures from auditors are stored in the directory specified in the exchange configuration section “exchangedb” under the option “AUDITOR_BASE_DIR”. The exchange does not care about the specific names of the files in this directory.

Each file must contain a header with the public key information of the auditor, the master public key of the exchange, and the number of signed denomination keys:

```
struct AuditorFileHeaderP {
  struct TALER_AuditorPublicKeyP apub;
  struct TALER_MasterPublicKeyP mpub;
  uint32_t dki_len;
};
```

This is then followed by dki_len signatures of the auditor of type struct TALER_AuditorSignatureP, which are then followed by another dki_len blocks of type struct TALER_DenominationKeyValidityPS. The auditor’s signatures must be signatures over the information of the corresponding denomination key validity structures embedded in a struct TALER_ExchangeKeyValidityPS structure using the TALER_SIGNATURE_AUDITOR_EXCHANGE_KEYS purpose.

APPENDIX

A

GNU FREE DOCUMENTATION LICENSE

Version 1.3, 3 November 2008

A.1 0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

A.2 1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

A.3 2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

A.4 3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

A.5 4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various partiesâ€"for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

A.6 5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

A.7 6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

A.8 7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

A.9 8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

A.10 9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

A.11 10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See https://www.gnu.org/licenses/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

A.12 11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

A.12.1 ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
Copyright (C)  YEAR  YOUR NAME.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.3
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.
A copy of the license is included in the section entitled "GNU
Free Documentation License".
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with â€¦ Texts." line with this:

```
with the Invariant Sections being LIST THEIR TITLES, with the
Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.