Setting up a GNU Taler Auditor

Version 1.0 from August 4, 2021

Code Blau GmbH
Klemkestr. 39
13409 Berlin
Abstract

Code Blau has setup an instance of the GNU Taler Auditor software for a test-environment. This is the log of the taken steps and the resulting audit report.

contact: Code Blau GmbH
email: contact@codeblau.de
fon: +49.30.65004524
fax: +49.30.55145804
address: Klemkestr. 39
city: 13409 Berlin
url: http://www.codeblau.de
1 Introduction

In June and July 2021, Code Blau has setup an instance of the GNU Taler Auditor system to audit the GNU Taler Exchange operated by Taler Systems SA, at https://exchange.chf.taler.net.

The preparation and setup was done by Özgür Kesim from Code Blau GmbH in cooperation with Christian Grothoff from Taler Systems SA.

2 Setup steps

We follow the instructions given in section 11, GNU Taler Auditor Operator Manual of the GNU Taler documentation

2.1 Preparation Debian 10

After installing Debian 10 on a server, we follow the instructions on 11.2.2. and prepare the package manager apt for the use of download-URL's provided by Taler Systems SA.

After a call to apt update we install the GNU Taler packages for the auditor and postgres with a call to apt install taler-auditor postgres-11.

2.2 Creating an offline signing-key

On a separate, isolated system, also with package taler-auditor installed, the offline key for signing denominations of the exchange is created:

```
root@offline# taler-config -s auditor -o BASE_URL -V https://bfh.auditor.codeblau.de
root@offline# taler-config -s exchange -o MASTER_PUBLIC_KEY -V DF0DFM8BRBAFCYGC4F6KBZTX5JR19K8YJ8P2AQT39ZJ629HSZN0
root@offline# taler-auditor-offline setup
EWY86W5DGQP80YKY72R4J0NQE7K9JNIDNMFYNS8DPAJ28MKXG
```

The highlighted result is the freshly generated public key that is than sent to the exchange operator over a secure communication channel.

The signing system is then taken offline.

2.3 Initial setup

The actual auditor system needs to be prepared, too:

```
root@auditor# taler-config -s auditor -o BASE_URL -V https://bfh.auditor.codeblau.de
```
2.4 Technical user accounts

We continue with section 11.2.3 to create technical users for various operational tasks:

**ingress** – maintains a network connection to the exchange and replicates the database of the exchange into the database taler-ingress.

**talersync** – has read-access to the replica of the exchange database and maintains a copy of that replica with a locally defined schema into the database talersync.

**auditor** – has read-access to the local copy of the exchange database and performs the actual evaluation/auditing of the contents.

```bash
root@auditor:/etc/taler-auditor# adduser --disabled-password auditor
root@auditor:/etc/taler-auditor# adduser --disabled-password ingress
root@auditor:/etc/taler-auditor# adduser --disabled-password talersync

root@auditor:/etc/taler-auditor# su - postgres
postgres@auditor:~$ createuser auditor
postgres@auditor:~$ createuser ingress
postgres@auditor:~$ createuser talersync
postgres@auditor:~$ createdb -O ingress taler-ingress
postgres@auditor:~$ createdb -O talersync taler-sync
postgres@auditor:~$ createdb -O auditor taler-auditor
```

2.5 Database creation

We then create the databases taler-ingress for the replication of the exchange database, taler-sync for an internal copy of the replica, both with appropriate rights to the appropriate technical users.

```bash
root@auditor:~# su - ingress
ingress@auditor:~$ psql -d taler-ingress
taler-ingress=> GRANT SELECT ON ALL TABLES IN SCHEMA public TO talersync;

root@auditor:~# su - talersync
talersync@auditor:~$ psql -d taler-sync
taler-sync=> GRANT SELECT ON ALL TABLES IN SCHEMA public TO auditor;

auditor@auditor:~$ taler-config -s auditordb-postgres -o CONFIG -V postgres:///taler-auditor
auditor@auditor:~$ taler-auditor-dbinit
```

2.6 Setting up a tunnel to the exchange

We create a SSH keypair for the ingress user and send the public key to the exchange operator.

```bash
ingress@auditor:~$ cat .ssh/id_ed25519.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIPYIhuWnJgSmiCMjoPX/NlkAAMMYFvAzmRMCuD25GUQQ ingress@codeblau.de
```
After sending the public key to the exchange operator, we create a SSH-tunnel to the exchange and use the port-forwarding feature of ssh in order to access the postgres instance at the exchange.

```
ingress@auditor:~$ ssh egress@exchange.chf.taler.net -L5555:localhost:5432
```

## 2.7 Exchange database replication

With the provided tunnel to the exchange, we can now setup the replication of the exchange database.

```
postgres@auditor:~$ psql -d taler-ingress
psql (13.3 (Debian 13.3-1), server 11.12 (Debian 11.12-0+deb10u1))
Type "help" for help.

taler-ingress=# CREATE SUBSCRIPTION bhf CONNECTION
            'dbname=taler-exchange host=localhost user=egress password=XXXXXXX port=5555'
        PUBLICATION codeblau;
NOTICE: created replication slot "bhf" on publisher
CREATE SUBSCRIPTION
```

On success, this provides us with a replica of the exchange database:

```
taler-ingress=# \d
List of relations
<table>
<thead>
<tr>
<th>Schema</th>
<th>Name</th>
<th>Type</th>
<th>Owner</th>
</tr>
</thead>
<tbody>
<tr>
<td>public</td>
<td>aggregation_tracking</td>
<td>table</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>aggregation_tracking_aggregation_serial_id_seq</td>
<td>sequence</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>auditor_denom_sigs</td>
<td>table</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>auditor_denom_sigs_auditor_denom_serial_seq</td>
<td>sequence</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>auditors</td>
<td>table</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>auditors_auditor_userid_seq</td>
<td>sequence</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>denomination_revocations</td>
<td>table</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>denomination_revocations_denomination_serial_id_seq</td>
<td>sequence</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>denominations</td>
<td>table</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>denominations_denominations_serial_seq</td>
<td>sequence</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>deposits</td>
<td>table</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>deposits_deposit_serial_id_seq</td>
<td>sequence</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>exchange_sign_keys</td>
<td>table</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>exchange_sign_keys_esk_serial_seq</td>
<td>sequence</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>known_coins</td>
<td>table</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>known_coins_known_coin_id_seq</td>
<td>sequence</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>prewire</td>
<td>table</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>prewire_prewire_userid_seq</td>
<td>sequence</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>recoup</td>
<td>table</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>recoup_recoup_userid_seq</td>
<td>sequence</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>recoup_refresh</td>
<td>table</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>recoup_refresh_recoup_refresh Unsure</td>
<td>sequence</td>
<td>ingress</td>
</tr>
<tr>
<td>public</td>
<td>refresh_commitments</td>
<td>table</td>
<td>ingress</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

(47 rows)
And data is actually arriving:

taler-ingress=# SELECT * from auditors;

| auditor_uuid | auditor_pub | auditor_name | ... |
|--------------+-------------|--------------+-----|
| c70a19091b2172ff0cc40f93a19a97db84704444d92b4905e476eadf0dedda2 | BFH Auditor | ... |
| 773c8370ad81af6403d3f1c5824815bb8f34aca16be47faa5436caa87e8a73b | Code Blau GmbH | ... |

(2 rows)

2.8 Copy of the Replica

After the creation of the replica, we need to grant access to the table.

ingress@auditor:~$ psql -d taler-ingress

psql (13.3 (Debian 13.3-1), server 11.12 (Debian 11.12-0+deb10u1))
Type "help" for help.

ingress@auditor:~$ GRANT SELECT ON ALL TABLES IN SCHEMA public TO talersync;

GRANT

We can now create a controlled copy of the ingress database:

talersync@Auditor:~$ taler-config -c .config/src.conf -s exchange -o DB -V "postgres"
talersync@Auditor:~$ taler-config -c .config/src.conf -s exchangedb-postgres -o CONFIG -V "postgres:///taler-ingress"
talersync@Auditor:~$ taler-config -c .config/dst.conf -s exchange -o DB -V "postgres"
talersync@Auditor:~$ taler-config -c .config/dst.conf -s exchangedb-postgres -o CONFIG -V "postgres:///taler-sync"
talersync@Auditor:~$ taler-exchange-dbinit -c .config/dst.conf
talersync@Auditor:~$ taler-auditor-sync -s .config/src.conf -d .config/dst.conf -t

2.9 Exchange configuration

Following section 11.5.1, we setup the auditor with the necessary parameters provided by the exchange.

auditor@Auditor:~$ taler-auditor-exchange -u https://chf.exchange.taler.net/
         -m df0dfm8brbafcygcf4e6kbztsxj19k0YJ0P2AQ739ZJ629HSZN0
auditor@Auditor:~$ taler-config -s exchange -o master_public_key
         -v df0dfm8brbafcygcf4e6kbztsxj19k0YJ0P2AQ739ZJ629HSZN0
auditor@Auditor:~$ taler-config -s base_url -v https://exchange.chf.taler.net/

2.10 Signing and uploading denominations

We bring the signing system online again and prepare it for the signing operation with the same commands as in 2.9:

root@offline:~$ taler-auditor-exchange -u https://chf.exchange.taler.net/
         -m df0dfm8brbafcygcf4e6kbztsxj19k0YJ0P2AQ739ZJ629HSZN0
root@offline:~$ taler-config -s exchange -o master_public_key
         -v df0dfm8brbafcygcf4e6kbztsxj19k0YJ0P2AQ739ZJ629HSZN0
root@offline:~$ taler-config -s base_url -v https://exchange.chf.taler.net/

We then download the denomination public keys from the exchange, sign...
and upload them again:

```bash
root@offline:~$ taler-auditor-offline -L debug download > input.json
Jun 22 17:13:49-425973 taler-auditor-offline-15095
  INFO Received keys from URL `https://exchange.chf.taler.net/keys' with status 200.
Jun 22 17:13:49-559136 taler-auditor-offline-15095
  INFO Successfully downloaded exchange's keys
Jun 22 17:13:49-559222 taler-auditor-offline-15095
  INFO Connecting to auditor at URL `https://auditor.chf.taler.net/service/' (0x55e00a1b8d20).
Jun 22 17:13:49-562612 taler-auditor-offline-15095
  INFO Disconnecting from auditor at URL `https://auditor.chf.taler.net/service/' (0x55e00a1b8d20)
root@offline:~# taler-auditor-offline sign < input.json > output.json
root@offline:~# taler-auditor-offline upload < output.json
```

After these calls, the signing system is taken offline again. The auditor has now successfully signed the denominations of the exchange, as can be verified with the data provided by the URL `https://exchange.chf.taler.net/keys'.

### 2.11 Generating an audit report

The setup is now complete for the auditor account to run the actual evaluation/auditing of the exchange:

```bash
auditor@auditor:~$ taler-auditor
```

This results in a report as pictured in appendix A.
A The produced auditor report

Table Auditor Report

<table>
<thead>
<tr>
<th>Auditor</th>
<th>Code</th>
<th>August 4, 2021</th>
</tr>
</thead>
</table>

This report is based on a single report rendered by the Alten GmbH Public Auditing, where the report is prepared and the number of the report is available in the report header.

The report is rendered in the following format:

4 Major irregularities

This section describes the possible major irregularities on the side of the exchange and, therefore, on the side of the auditors. These irregularities are sorted according to their severity and described in the following order:

4.1 Lag

Lag constitutes a serious matter when there are delays in the exchange's public audit. This section is concerned with the exchange's public audit, which is necessary for the exchange to continue trading. The delay in the exchange's public audit is the only reason why the exchange's public audit is not yet processed.

4.2 Arithmetic problems

Arithmetic problems are common in the exchange's public audit. This section is concerned with the exchange's public audit, which is necessary for the exchange to continue trading. The arithmetic problems are the only reason why the exchange's public audit is not yet processed.

4.3 Outgoing wire transfer subject issues

Outgoing wire transfer subject issues are common in the exchange's public audit. This section is concerned with the exchange's public audit, which is necessary for the exchange to continue trading. The outgoing wire transfer subject issues are the only reason why the exchange's public audit is not yet processed.

4.4 Outgoing wire transfer timestamp issues

Outgoing wire transfer timestamp issues are common in the exchange's public audit. This section is concerned with the exchange's public audit, which is necessary for the exchange to continue trading. The outgoing wire transfer timestamp issues are the only reason why the exchange's public audit is not yet processed.

4.5 Denomination key invalid at time of withdrawal

Denomination key invalid at time of withdrawal is common in the exchange's public audit. This section is concerned with the exchange's public audit, which is necessary for the exchange to continue trading. The denomination key invalid at time of withdrawal is the only reason why the exchange's public audit is not yet processed.

4.6 Other issues

Other issues are common in the exchange's public audit. This section is concerned with the exchange's public audit, which is necessary for the exchange to continue trading. The other issues are the only reason why the exchange's public audit is not yet processed.

5 Log

This section describes the log entries that have been made during the exchange's public audit. The log entries are necessary for the exchange to continue trading.

5.1 Transactions with unidentified subject

This section describes the transactions with unidentified subject that have been made during the exchange's public audit. The transactions with unidentified subject are necessary for the exchange to continue trading.

5.2 Coast-to-coast balance

This section describes the coast-to-coast balance that has been made during the exchange's public audit. The coast-to-coast balance is necessary for the exchange to continue trading.

5.3 Wire transfer deduction

This section describes the wire transfer deduction that has been made during the exchange's public audit. The wire transfer deduction is necessary for the exchange to continue trading.

5.4 Other issues

This section describes the other issues that have been made during the exchange's public audit. The other issues are necessary for the exchange to continue trading.

6 Delays and handling

This section describes the delays and handling that have been made during the exchange's public audit. The delays and handling are necessary for the exchange to continue trading.

6.1 Delays and handling

This section describes the delays and handling that have been made during the exchange's public audit. The delays and handling are necessary for the exchange to continue trading.

6.2 Handling of discrepancies

This section describes the handling of discrepancies that have been made during the exchange's public audit. The handling of discrepancies is necessary for the exchange to continue trading.